Overhead analysis of universal concatenated quantum codes

Christopher Chamberland,[1,*] Tomas Jochym-O'Connor,[1,2,†] and Raymond Laflamme[1,3,4]

[1]*Institute for Quantum Computing and Department of Physics and Astronomy, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1*

[2]*Walter Burke Institute for Theoretical Physics and Institute for Quantum Information & Matter, California Institute of Technology, Pasadena, California 91125, USA*

[3]*Perimeter Institute, Waterloo, Ontario, Canada N2L 2Y5*

[4]*Canadian Institute For Advanced Research, Toronto, Ontario, Canada M5G 1Z8*

(Received 27 September 2016; published 8 February 2017; corrected 15 February 2017)

We analyze the resource overhead of recently proposed methods for universal fault-tolerant quantum computation using concatenated codes. Namely, we examine the concatenation of the 7-qubit Steane code with the 15-qubit Reed-Muller code, which allows for the construction of the 49- and 105-qubit codes that do not require the need for magic state distillation for universality. We compute a lower bound for the adversarial noise threshold of the 105-qubit code and find it to be 8.33×10^{-6}. We obtain a depolarizing noise threshold for the 49-qubit code of 9.69×10^{-4} which is competitive with the 105-qubit threshold result of 1.28×10^{-3}. We then provide lower bounds on the resource requirements of the 49- and 105-qubit codes and compare them with the surface code implementation of a logical T gate using magic state distillation. For the sampled input error rates and noise model, we find that the surface code achieves a smaller overhead compared to our concatenated schemes.

DOI: 10.1103/PhysRevA.95.022313

I. INTRODUCTION

Fault-tolerant computations provide a means to control and suppress error rates to arbitrarily low levels, without a detrimental overhead in terms of the number of qubits and computation time. However, estimating the additional resources that would be required for such computations is an important area of study as physical architectures begin to approach the realms of scalability [1–12]. As such, it has become increasingly important to evaluate whether different architectures have particular advantages over one another with respect to targeting the implementation of a given algorithm with required accuracy.

The early proposals for fault-tolerant architectures were given by concatenated quantum error correcting codes, where qubits forming an error correcting code are reencoded for further protection. This provides a means to reduce the error rate in a double-exponential manner as the number of concatenation levels increase, assuming the physical error rate is below some threshold value, deemed the fault-tolerance threshold [13,14]. Subsequently, topological quantum codes were proposed, beginning with the surface code [15]. There, logical information is stored in highly nonlocal degrees of freedom, while using local stabilizer checks, thus providing the ability to increase the protection of the code by increasing the size of the physical lattice encoding the information. One of the primary advantages of schemes such as the surface code is its high threshold value in comparison to concatenated code schemes. For depolarizing noise on each gate and memory location, the threshold value of the surface code is on the order of 10^{-2} [16,17] in comparison to 10^{-3} [18,19] for most concatenated schemes. Moreover, stabilizer syndrome checks are simpler as the weight of the checks remains fixed as the distance increases, unlike in the case of concatenated architectures.

The goal of this work is to estimate the overhead with a recently proposed scheme for fault-tolerant computation using concatenated codes that allows for the implementation of a universal set of gates without the need for magic state distillation [20,21]. Magic state distillation forecasts to be a challenge for logical computation on 2D topological codes, such as the surface codes. As such, many recent developments in the area of quantum error correction have focused on circumventing no-go theorems regarding the implementation of universal quantum logic using transversal gates (the simplest form of fault-tolerant operation) [22–24]. The 105-qubit scheme circumvents these no-go theorems by concatenating complementary sets of transversal gates and lead to a fault-tolerant threshold of 1.28×10^{-3} for depolarizing noise. While this threshold rate compares favorably with other concatenated methods, it is still an order of magnitude below that of the surface code, thus will require higher distance iterations to reach a given target error rate when compared to the surface code. However, the primary advantage of the universal scheme is to avoid magic state distillation and as such this potential reduction in complexity would allow for the concatenated model to have a reduced overall overhead. The main result of this work is to provide a lower bound on the number of qubits and gates that would be required for the universal concatenated scheme to reach particular target error rates, given a physical depolarizing error rate. In order to do so, the overhead in state preparation needed for Steane error correction [25] as well as the suppression of the logical error rate as a function of concatenation level are determined.

In this work, we make no assumptions on the locality of the code, treating each location and gate with equal weighting in terms of accessibility and error probability. We acknowledge that this may be unrealistic for many current realizations of physical quantum computing experiments, although there are some exceptions [26]; however, in order to asses whether such

*c6chambe@uwaterloo.ca
†trjochym@uwaterloo.ca

a scheme would provide a benefit in terms of overhead with respect to local codes such as the surface code, it is necessary to treat them on equal footing. If a nonlocal scheme were to show significant improvements over schemes that are local, then this would motivate the experimental community to optimize for better performance of long-range gates. However, if codes such as the surface code are shown to be more efficient in even this nonrealistic nonlocal error model, then this would further solidify the status of such local codes as the most promising physical schemes.

In Sec. II we review the concepts behind Steane error correction and outline important parameters for the counting of resources related to the logical ancilla state preparation. In Sec. III we review the concepts of malignant set counting [18] which allows for the establishment of a lower bound for the theoretical threshold. In Sec. IV we review the 105-qubit code that allows for universal fault-tolerant computation, and discuss a reduction of this code to 49 qubits [27]. We additionally present models for decoding in these codes given a particular error syndrome, highlighting important differences needed depending on the logical gate implemented. Additionally, we provide new logical ancilla state preparation circuits in order to reduce circuit depth allowing for increased success probability and higher threshold values and we present the adversarial noise threshold. In Sec. V we review and present the depolarizing noise threshold results for the 105- and 49-qubit codes, respectively. In Sec. VI we calculate a lower bound on the qubit and gate resource overhead for the implementation of the Hadamard and controlled-NOT (CNOT) gates under depolarizing noise as they dominate the resource requirements in the universal concatenated method. We compare these results with an estimate for the surface code qubit resource overhead using magic state distillation. We conclude the article with a review of the results and their impact for quantum architectures, and provide open questions and targets for future universal quantum codes.

II. FAULT-TOLERANT SCHEME FOR ERROR CORRECTION

In this section, we describe the fault-tolerant error correction scheme used in the implementation of the universal concatenated quantum codes considered in this paper. We use Steane's method for fault-tolerant error correction [18,25] which applies to Calderbank-Shor-Steane (CSS) codes [28] with stabilizer generators given by tensor products of all X and I operators (X generators) or tensor products of all Z and I operators (Z generators).

In Steane's method, the ancilla qubits used for syndrome extraction are encoded using the same CSS code that protects the data qubits. Since the stabilizer generators are separated into X and Z generators, we can measure them separately. To measure the Z generators, we prepare the ancilla in the encoded state $|\overline{+}\rangle = (|\overline{0}\rangle + |\overline{1}\rangle)/\sqrt{2}$ and apply transversal CNOT gates with the data block as control and the ancilla block as target. This will propagate each X error in the data block to the corresponding position in the ancilla block. The ancilla is encoded in the $|\overline{+}\rangle$ state since it is an eigenstate with eigenvalue one of the logical X Pauli operator and thus the CNOT gate has no effect on the encoded state of the ancilla. If no errors occur in either the preparation of $|\overline{+}\rangle$ or the encoded CNOT gate, then measuring the ancilla block in the Z basis detects the X errors when considering the Z stabilizers of the code which are products of the individual measurements. Similarly, to detect and correct Z errors, we prepare an ancilla state in the $|\overline{0}\rangle$ basis and apply transversal CNOT gates with the data block as target and ancilla block as control. The sequence ends by performing a measurement of the ancilla in the X basis. In general, the circuits used for encoding the $|\overline{+}\rangle$ and $|\overline{0}\rangle$ ancilla states are not fault tolerant since a single fault can propagate badly, leading to a high weight error at the output of the encoding circuit. In order to ensure fault tolerance in Steane's error correction method, a verification step is needed to detect if errors occurred during the encoding of the ancilla states. For the $|\overline{0}\rangle$ state, X errors can propagate from the ancilla to the data. Consequently, we encode a "verifier" state in the $|\overline{0}\rangle$ state and apply an encoded CNOT gate with the ancilla block as control and the verifier block as target. Any X errors are then detected by performing a measurement in the Z basis. Not only are X errors detected, but after a classical error correction step the eigenvalue of the encoded $\overline{Z}$ operator is also found. If a nontrivial syndrome is measured or the eigenvalue of $\overline{Z}$ is found to be -1, all ancilla blocks are rejected and the error correction protocol starts over. The latter will play an important role when considering the resource overhead of the concatenated 49- and 105-qubit code.

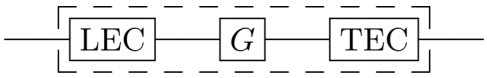

FIG. 1. Illustration of an extended rectangle consisting of a gate G along with its leading and trailing error correction circuit. Extended rectangles are used in the fault-tolerant implementation of a logical gate. The leading and trailing error correction circuits are implemented using Steane's method (see Sec. II) which is a protocol for a fault-tolerant error correction.

III. MALIGNANT SET COUNTING OVERVIEW

Following [18], we define a k-rec to be the encoded gate under consideration acting on codewords followed by a trailing error correction (TEC) circuit at the kth level of concatenation. A k-exrec (where exrec stands for *extended rectangle*) corresponds to a leading error correction (LEC) circuit followed by a k-rec (see Fig. 1).

Consider an arbitrary 1-exrec. A location can be either a 0-preparation, 0-measurement, or a gate (note that the identity gate corresponds to a resting qubit). We say that a set of m locations is *benign* if the 1-rec contained in the 1-exrec is *correct* for arbitrary faults occurring at those locations. Otherwise, the set of locations is defined to be *malignant*. A 1-rec is correct if it takes any input with no more than one error per block to an output with no more than one error per block.

The idea behind malignant set counting is that for a fixed number of locations in the circuit, we would like to count all such sets of locations which are malignant. A 1-exrec will be *bad* if it contains faults at a malignant set of locations, otherwise we will define it to be *good*. We can generalize the

definition of goodness and badness to a k-exrec (for $k > 1$) by defining the k-exrec to be bad if it contains independent bad $(k-1)$-exrecs at a malignant set of locations, otherwise we define it to be good.

In this section, we will consider an independent stochastic noise model where fault locations are independently and identically distributed. The operations at the chosen locations are arbitrary trace-preserving completely positive maps. The latter noise model is also known as adversarial noise. Since an arbitrary fault can be expanded in terms of Pauli operators, a set of locations will be benign if the 1-rec contained in the 1-exrec is correct for all Pauli faults at those locations. Furthermore, if all of the locations are in the LEC of the 1-exrec, then by definition the 1-rec contained in the 1-exrec is correct. Therefore, when counting malignant sets of locations, we will exclude the cases where all faults are in a LEC. As in Ref. [18], for a fixed set of locations, we insert all combinations of X and Z Pauli faults at those locations and propagate them through the 1-exrec. If for all combinations of Pauli faults the 1-rec contained in the 1-exrec is correct, then the set of locations is benign.

Consider a 1-exrec with a total of L locations and suppose that the ECs have L_{EC} locations. For a circuit simulating a t-qubit gate, we define

$$L_{n,t} \equiv \binom{L}{n} - t\binom{L_{\mathrm{EC}}}{n}, \tag{1}$$

where n is the number of faulty locations in the circuit. Similarly, define

$$A_{n,t} \equiv f_n L_{n,t}, \tag{2}$$

where $f_n = n_{mal}/N$ is the fraction of malignant locations containing n faults. Due to limits in computation time, when calculating the noise threshold for a particular gate, we count malignant sets of locations for up to m faults and assume that all $m+1$ sets of locations are malignant. For large enough m, the error in the truncation can be made very small [18]. Defining the probability of a k-exrec to be bad by $\varepsilon^{(k)}$, we can use the statistical independence of bad $(k-1)$-exrecs to calculate an upper bound on $\varepsilon^{(k)}$:

$$\begin{aligned}\varepsilon^{(k)} \leqslant\ & A_{2,t}(\varepsilon^{(k-1)})^2 + A_{3,t}(\varepsilon^{(k-1)})^3 + \cdots \\ & + A_{m-1,t}(\varepsilon^{(k-1)})^{m-1} + L_{m,t}(\varepsilon^{(k-1)})^m.\end{aligned} \tag{3}$$

Note that in the above equation we have assumed that all level-0 locations in our circuit have the same fault rate ε. In our scheme, noisy circuits will be constructed from $|0\rangle$ and $|+\rangle$ initialization, Hadamard and CNOT gates, as well as single-qubit measurements in the X and Z eigenbases. The level-0 locations used in our circuits are given in Table I. Since the wait time of a qubit during the application of a gate could differ from the wait time of a measurement, we can associate different failure rates for the two rest cycles. Generalizing Eq. (3) to include different failure rates for distinct level-0 locations, we obtain

TABLE I. Types of level-0 locations present in the considered 1-exrec's.

Level-0 location label	Types of level-0 locations
1	Rest during a gate cycle
2	Rest during a measurement cycle
3	Preparation of $\|0\rangle$
4	Preparation of $\|+\rangle$
5	Measurement of X
6	Measurement of Z
7	CNOT gate
8	H gate
9	T gate

$$\begin{aligned}\varepsilon^{(k)} \leqslant\ & \sum_{j\leqslant i=1}^{l_{\max}} \alpha^{(2)}_{ij,t}\varepsilon_i^{(k-1)}\varepsilon_j^{(k-1)} \\ & + \sum_{l\leqslant j\leqslant i=1}^{l_{\max}} \alpha^{(3)}_{ijl,t}\varepsilon_i^{(k-1)}\varepsilon_j^{(k-1)}\varepsilon_l^{(k-1)} + \cdots \\ & + L_{m,t}(\varepsilon^{(k-1)})^m,\end{aligned} \tag{4}$$

where $l_{\max}$ depends on the types of locations in a particular exrec. Hence, exrecs excluding H or T gates would have $l_{\max} = 7$. The indices correspond to a particular level-0 location and are summed over all location types of the exrec under consideration. $\alpha^{(2)}_{ij,t}$ corresponds to the number of malignant pairs of types i and j for a gadget acting on t qubits. For example, following the indexing from Table I, 5 is the label used for a measurement in the X basis and 7 is the label for a CNOT gate. Therefore, $\alpha^{(2)}_{57,t}$ corresponds to the number of malignant pairs in the exrec where one location is a measurement in the X basis and the other location is a CNOT gate. Generalizing to larger sets of locations, $\alpha^{(n)}_{i_1 i_2 \ldots i_n,t}$ is the number of malignant sets of n locations of type $i_1, i_2, \ldots, i_n$. For the remainder of this section, we will assume that all location types have the same failure probability, so that the upper bound in Eq. (3) will be used for the threshold calculation (in this case $A_{2,t}$ is simply the sum of all the elements of the $\alpha^{(2)}_t$ matrix, and $\varepsilon_i^{(k-1)} = \varepsilon_j^{(k-1)} = \varepsilon^{(k-1)}$).

From Eq. (3), it follows that

$$\varepsilon^{(k)} \leqslant A'_t(\varepsilon^{(k-1)})^2, \tag{5}$$

where the threshold estimate is

$$\varepsilon \leqslant \varepsilon_0 = (A'_t)^{-1}. \tag{6}$$

A'_t can be calculated from the polynomial equation

$$\begin{aligned}& (A'_t)^m - A_{2,t}(A'_t)^{m-1} - A_{3,t}(A'_t)^{m-2} - \cdots \\ & - A_{m-1,t}A'_t - L_{m,t} = 0.\end{aligned} \tag{7}$$

Since the coefficients $A_{j,t}$ are strictly increasing for increasing j, there will only be one positive solution to Eq. (7).

Recall that the ancilla blocks used to extract the error syndrome need to successfully pass a verification test, otherwise the ancillas are rejected and the computation is started over. Instead, the failure probability for a k-exrec should be upper

bounded conditioned on the acceptance of all ancilla blocks. In calculating the coefficients $A_{j,t}$, we count sets of locations such that faults at those locations cause the k-exrec to fail but lead to acceptance of all ancilla blocks. Hence, the upper bound in Eq. (3) actually corresponds to the *joint* probability of acceptance of all ancilla blocks and failure of the k-exrec. We can use Bayes' rule to obtain the conditional probability of failure given the acceptance of all ancillas. Using the notation from Ref. [18], we need to calculate the probability $P^{(k)}_{|\overline{0}\rangle,\text{accept}}$ and $P^{(k)}_{|\overline{+}\rangle,\text{accept}}$ that a level-k encoded $|\overline{0}\rangle$ or $|\overline{+}\rangle$ passes the verification test. Distinction between $P^{(k)}_{|\overline{0}\rangle,\text{accept}}$ and $P^{(k)}_{|\overline{+}\rangle,\text{accept}}$ is important since the encoding circuits for $|\overline{0}\rangle$ and $|\overline{+}\rangle$ are not symmetric for the 15-qubit Reed-Muller code (since the code is not self-dual) and so they will contain a different number of locations. We define

$$P^{(k)}_{\text{min,accept}} \equiv \min\left\{P^{(k)}_{|\overline{0}\rangle,\text{accept}}, P^{(k)}_{|\overline{+}\rangle,\text{accept}}\right\}, \tag{8}$$

which corresponds to the smallest of the two acceptance probabilities. In Steane's error correction protocol, there will always be an equal number of encoded $|\overline{0}\rangle$ and $|\overline{+}\rangle$ circuits in the k-EC's. We define n_{anc} to be the number of encoded $|\overline{0}\rangle$ or $|\overline{+}\rangle$ circuits in the k-exrec under consideration. Using Bayes' rule, the probability of failure $\varepsilon^{(k)}$ for the k-exrec, conditioned on acceptance of all ancillas, can be upper bounded as

$$\varepsilon^{(k)} \leqslant \left(P^{(k)}_{\text{min,accept}}\right)^{-n_{anc}} \varepsilon^{(k)}_{\text{joint}}, \tag{9}$$

where $\varepsilon^{(k)}_{\text{joint}}$ is upper bounded by Eq. (3). Let $C_{|\overline{0}\rangle}$ and $C_{|\overline{+}\rangle}$ correspond to the number of locations in the encoding and verification circuits of $|\overline{0}\rangle$ and $|\overline{+}\rangle$ in the ECs. For an ancilla to be rejected, the previous encoding and verification circuits must contain at least one bad $(k-1)$-exrec. Therefore, a lower bound on $P_{\text{min,accept}}$ is given by

$$P^{(k)}_{\text{min,accept}} \geqslant 1 - C_{\max}\varepsilon^{(k-1)}, \tag{10}$$

where $C_{\max} \equiv \max\{C_{|\overline{0}\rangle,|\overline{+}\rangle}\}$. Using (8)–(10) and assuming that $\varepsilon^{(k-1)} < (A'_t)^{-1}$, we have that

$$\varepsilon^{(k)} \leqslant \left(1 - \frac{C_{\max}}{A'_t}\right)^{-n_{anc}} A'_t(\varepsilon^{(k-1)})^2. \tag{11}$$

Defining

$$A''_t \equiv \left(1 - \frac{C_{\max}}{A'_t}\right)^{-n_{anc}} A'_t, \tag{12}$$

the threshold estimate is then given by

$$\varepsilon_0 = (A''_t)^{-1}. \tag{13}$$

IV. THE 105- AND 49-QUBIT CODES

In this section we will review the construction of the 105-qubit concatenated code that allows for universal fault-tolerant logical gate implementations, as well as a simplification of this construction to 49 qubits [27]. We shall then discuss the different ways we implement decoding and error correction given syndrome measurements. Our methods for constructing the ancilla state preparation circuits can be found in the Appendix.

A. Fault-tolerant universal concatenated quantum codes

The idea behind the 105-qubit construction is to use two different error correcting codes, with different sets of transversal gates (logical gates that can be implemented by applying individual gates to each qubit composing the code), in concatenation in order to implement a universal set of fault-tolerant logical operations [20]. In the construction of the 105-qubit code, the outer code is designated as the 7-qubit code and therefore contains transversal Clifford operations. The inner code, that is, each qubit composing the 7-qubit code, is the 15-qubit code which has transversal logical CNOT and T gates, where $T = |0\rangle\langle 0| + e^{i\pi/4}|1\rangle\langle 1|$ and completes the universal gate set when combined with Clifford operations. Since the T gate cannot be implemented transversally for the 7-qubit code, any logical construction will necessarily have to couple different qubits in the code (in the case of the 105-qubit code, coupling qubits correspond to coupling blocks of qubits composing the code). There exists a construction for the implementation of the T gate using a sequence of CNOT gates and a T gate on the qubits of the 7-qubit code. However, since each of these operations is implemented at the logical level from the perspective of the 15-qubit code, they will all be transversal with respect to this code. Therefore, any single-qubit fault occurring during the action of this sequence of gates may lead to a propagation of the faults, but in a controlled manner to only a single location to each of the code blocks. Therefore, the logical gate remains fault tolerant as any single fault remains correctable.

The logical Hadamard is implemented by applying the logical Hadamard gate on each of the seven encoded code blocks (as the Hadamard is transversal for the 7-qubit code). However, the Hadamard is not transversal on each 15-qubit code block, and as such a single error may spread to form a logical fault on an individual code block. However, since only one code block is corrupted, overall the error will remain correctable as such an error can be detected and corrected by the 7-qubit outer code syndrome. This complication will have important consequences for decoding, as explained further in this section.

An important observation is that the reencoding of the qubits into 15-qubit code blocks is only important to protect the blocks that have active gates in the implementation of the logical T gate. Therefore, it is only necessary to encode three code blocks, corresponding to the set of blocks that would correspond to a logical Pauli operator for the 7-qubit code [27]. Therefore, the total qubit count can be reduced to be $3 \times 15 + 4 = 49$ qubits. While the overall distance of the code will be lower, reduced from a distance 9 to distance 5 code, the logical operations will still be able to correct for arbitrary single-qubit faults, just as in the case of the 105-qubit code. This idea can be generalized further to any construction using as an outer code a 2D color code, which has transversal Clifford operations, and choosing a set of qubits that contain both logical Pauli operators and reencoding these qubits in a 3D color code containing the transversal CNOT and T gates required from the original scheme [29].

B. Decoding the 105-qubit code

As we have previously explained, the 105-qubit code is a distance 9 quantum error correcting code whose distance is sacrificed for the implementation of the non-globally-transversal H and T gates. However, the decoder should be designed such that for the CNOT gate any weight-4 error can be corrected. The 105-qubit syndromes consist of the 15-qubit syndromes on each of the code blocks composing the outer code, as well as the 6 syndromes of the 7-qubit code which correspond to syndromes across 4 blocks (since all stabilizers of the 7-qubit code have weight 4).

The most basic decoding scheme is the greedy decoder, where each of the 15-qubit code blocks is corrected according to their measured syndromes individually, and then the outer 7-qubit code is corrected independently according to the remaining syndrome.[1] The greedy decoder will fail to correct for all weight-4 errors. For example, consider the case when two 15-qubit code blocks each have a weight-2 Z error. Each code block is then corrected with the identified single-qubit recovery (since the 15-qubit code is weight-3 for Z errors, it can only correct weight-1 errors) resulting in a logical error on the two 15-qubit code blocks. Then, the outer 7-qubit stabilizers will identify this weight-2 logical fault with a logical fault on a third code block, and thus "correct" by implementing a logical Z on the third code block. As such, the final state will undergo a global logical Z error from the composition of the 3 logical code-block errors. Therefore, having corrected each of the 15-qubit code blocks, when correcting the 7-qubit code blocks, information from the 15-qubit syndromes will have to be used in the correction of the outer 7-qubit code.

The decoding of the 105-qubit code will be implemented using the following steps:

(1) Correct all of the 15-qubit code blocks individually, and store which code blocks underwent any correction.

(2) Update the 7-qubit syndromes according to the corrections from the 15-qubit code blocks.

(3) If the 7-qubit syndrome is trivial (no syndrome identifies an error), then correction complete.

(4) If the 7-qubit syndrome identifies an error on a code block that did not undergo a 15-qubit correction, and in addition there is a set of complementary blocks[2] that were corrected at the 15-qubit level, then perform further logical operations on the complementary blocks, then correction complete.

(5) Otherwise, correct the identified code block by applying a logical correction.

We illustrate the advantage of this decoding scheme by highlighting the example that the greedy decoder failed to correct. Consider weight-2 Z errors on the first and second code blocks. Each of these code blocks is corrected by applying a weight-1 correction, resulting in logical Z errors on each of the code blocks. Then, the 7-qubit syndrome would identify an error on code block 3 since the syndrome associated with Z_1Z_2 is equivalent to Z_3. Since the 15-qubit decoder did not make any corrections on code block 3, yet did make corrections to code blocks 1 and 2, which are complementary to code block 3, logical Z corrections are applied to each of these code blocks, therefore correcting all of the errors. One can verify that all weight-4 errors will be corrected by this scheme, thus achieving the promised distance of the code. Of course, for the implementation of the logical H and T gates, not all weight-4 errors will be corrected as the gates are not globally transversal. However, all weight-1 errors will still be corrected by our decoding scheme.

C. Decoding the 49-qubit code

The 49-qubit code sacrifices the full distance of the 105-qubit code by only encoding three code blocks, therefore reducing the overall distance to be 5. As such, any decoder will be able to correct at most any weight-2 error. The correction scheme implemented for the logical CNOT and T gates is as follows (as described below, a different decoder is used for the logical H):

(1) Correct the three 15-qubit code blocks, tracking which blocks contained errors.

(2) Update the 7-qubit syndromes according to the corrections from the 15-qubit code blocks.

(3) If the 7-qubit syndrome identifies a 15-qubit code block that had a trivial syndrome in step 1, then perform a weight-2 correction on the complementary single-qubit blocks, then correction complete.

(4) For any other nontrivial 7-qubit syndrome, correct according to the identified single or 15-qubit code block.

For the transversal CNOT, any weight-1 error will either be corrected originally by the 15-qubit syndrome measurement if it occurs on a code block or at the 7-qubit level if it occurs on an unencoded block. If two errors occur, there are four possibilities. If both errors occur on the same encoded code block, then they are corrected at the 15-qubit level, potentially resulting in a logical fault on that code block. However, such a logical fault will be detected by the 7-qubit stabilizers and as such will be corrected according to the protocol. If the two errors occur on different encoded 15-qubit code blocks, they will each be correctly identified at the 15-qubit level. If one error occurs on a code block and one on an unencoded block, then the error on the 15-qubit code block will be corrected, and only the error on the single qubit will remain when measuring the 7-qubit syndromes and will be correctly identified. The tricky case comes when errors occur on two different unencoded blocks, therefore not identified by the 15-qubit stabilizers. In this case, the code block that is complementary to these two errors will be identified when the 7-qubit stabilizers are measured, however, since no error had been identified on that code block at the 15-qubit level, the error complementary to that block is corrected on single qubits, resulting in an "error" that will be logically equivalent to the identity.

---

[1]Note that in Steane error correction, all of the syndromes are measured in parallel for both the 15-qubit code blocks as well as the outer 7-qubit code. Any correction made at the 15-qubit level that would modify the 7-qubit syndrome results can be accounted for in software.

[2]The complementary blocks are the two blocks that along with the identified block would form a logical error for the 7-qubit code. For example, $Z_1Z_2Z_3$ forms a logical error for the 7-qubit code, therefore, complementary blocks for block 1 are (2,3), yet also blocks (4,5) and (6,7), since they are logically equivalent.

In the case of the T gate, some of the distance is sacrificed in order to implement the nontransversal T gate, yet remains fault tolerant and can correct for any weight-1 error. Any weight-1 error is corrected equivalently to a Greedy decoder in the case of our protocol, and as such is corrected in the same manner as in the case of CNOT. The only difference in potential realizations is that a weight-1 error on any of the code blocks may spread to a weight-1 error on the other code blocks, yet they all remain correctable through the protocol by correcting at the 15-qubit level.

Unlike in the case of the 105-qubit code, it is important to point out that a different decoder must be used in the case of the logical Hadamard. Due to the non-fault-tolerant construction of the logical Hadamard on a given 15-qubit code block, a single fault on such a code block could result in a logical fault on that code block at the conclusion of the gate. Such an error would be corrected in the wrong fashion according to the above decoder, as it would go unrecognized by the 15-qubit stabilizers and would falsely identify a correction on the complementary single qubits. As such, the solution is to correct in a greedy fashion at the 7-qubit level, thus resulting in a correction on the single 15-qubit block. Therefore, the resulting protocol would be to correct in a greedy manner at the 15-qubit level and the 7-qubit level, and weight-2 errors on two single-qubit code blocks would therefore result in a logical fault unlike in the case of the transversal CNOT. As mentioned previously, not being able to correct all weight-2 errors is due to sacrificing the full code distance for the implementation of the logical H.

D. Adversarial noise threshold results for the 105-qubit code

We present the results of the threshold analysis for the adversarial noise model described in Sec. III. We will begin by computing the noise threshold for the Hadamard-exrec which turns out to be the circuit that provides a lower bound for the threshold of the 105-qubit code. From the analysis leading to Eq. (3), we first need to compute the coefficients $A_{j,t}$ where $j \in \{2,3,\ldots,m\}$ for some cutoff value m. Given the large number of locations in the exrecs for the 105-qubit code, it is computationally impractical to count all malignant sets of locations. In order to overcome this difficulty, we use a Monte Carlo method following the ideas of Ref. [30]. Suppose we want to count all malignant locations for a set of j faults, i.e., the coefficient $A_{j,t}$. Instead of looking at every combination of j locations within the full exrec, we can uniformly sample the set of all fault paths for a fixed set of j faults. We thus obtain an estimate of the fraction $\hat{f}_{j,t}$ of malignant faults for j sets of locations which in turn provides an estimate of the exact coefficient $A_{j,t}$. The standard error is obtained from the relation

$$\sigma_{j,t} = \sqrt{\hat{f}_{j,t}(1-\hat{f}_{j,t})/N}, \tag{14}$$

where N is the sample size. Due to computational limits, we set the cutoff value to $m = 6$ and choose a sample size of $N = 10^7$. Since the Hadamard-exrec only contains one LEC, $t = 1$ and from Table VIII

$$L_{n,1} = \binom{15067}{n} - \binom{7110}{n} \tag{15}$$

TABLE II. Value of Hadamard-exrec $A_{j,1}$ coefficients obtained from a Monte Carlo simulation which are used in the threshold calculation for adversarial noise.

$A_{j,1}$ coefficients for Hadamard 1-exrec	Monte Carlo estimate
$A_{2,1}$	$(6.80 \pm 0.24) \times 10^4$
$A_{3,1}$	$(1.73 \pm 0.03) \times 10^9$
$A_{4,1}$	$(1.30 \pm 0.05) \times 10^{13}$
$A_{5,1}$	$(6.44 \pm 0.20) \times 10^{16}$
$A_{6,1}$	$(2.42 \pm 0.06) \times 10^{20}$
$L_{7,1}$	3.47×10^{25}

The coefficients $A_{j,1}$ can be computed from

$$A_{j,1} = \hat{f}_{j,1} L_{j,1}. \tag{16}$$

Our Monte Carlo implementation was written in MATLAB. Solving Eq. (7) with the coefficients from Table II, we find that $A_1' = (8.92 \pm 0.23) \times 10^4$. Next, we need to compute $C_{\max}$ by counting the number of locations in the encoding and verification circuits of $|\bar{0}\rangle$ and $|\overline{+}\rangle$ in the ECs which can be found in the Appendix. The encoding and verification circuits contain four encoded state preparations ($|\bar{0}\rangle$ or $|\overline{+}\rangle$), three CNOT gates, and three measurement locations. Hence, we have that $C_{|\overline{+}\rangle} = 3254$ and $C_{|\bar{0}\rangle} = 3226$ so that $C_{\max} = 3254$. Since there are eight $|\overline{+}\rangle$ preparation circuits, then $n_{anc} = 8$. Using Eqs. (12) and (13), we calculate the noise threshold for the Hadamard-exrec to be

$$\varepsilon_0 = (8.33 \pm 0.28) \times 10^{-6}. \tag{17}$$

We can repeat the same calculations leading to Eq. (17) for the CNOT 1-exrec (which contains 28 545 locations compared to 15 067 locations for the Hadamard 1-exrec). For $N = 10^7$ iterations, we found that the fraction of malignant events $\hat{f}_{2,2}$ to $\hat{f}_{9,2}$ were all zero. Clearly, for a large enough number of iterations, some of these coefficients would be nonzero. However, given that the Hadamard-exrec circuit is more sensitive to input noise which will lower bound the threshold for the 105-qubit code, and due to limits in computation time, it is impractical to compute the coefficients $A_{j,2}$ for $N > 10^7$.

In an adversarial noise model calculation, locations fail independently and at random with the constraint that the error at each location is chosen by an adversary. The error could be correlated with errors at other failing locations. Although the threshold computed in Eq. (17) is quite low, it is important to keep in mind that malignant set counting is overly pessimistic for noise models such as depolarizing noise. As pointed out in [19], malignant set counting is independent of the underlying noise model and so ignores large quantities of information. For the remainder of this paper, we will focus on depolarizing noise and compute thresholds for the 49-qubit code. For the 105-qubit code, we will use the threshold results that were calculated in [21] using similar methods. The protocol for our simulation will be explained in Sec. V. Using the computed threshold results for both codes, we will obtain the resource overhead for performing encoded gates and compare the results with magic state distillation techniques applied to the surface code.

V. CONCATENATED 49- AND 105-QUBIT CODE THRESHOLDS

The goal of designing fault-tolerant architectures is to allow arbitrary long computations to be performed on large-scale quantum computers where the probability of failure can be made as small as desired. For concatenated coding schemes, fault-tolerant architectures give rise to *asymptotic thresholds* which corresponds to the error rate p_{th} such that for physical error rates $p < p_{th}$, the logical error rate can be made as small as desired for sufficiently large number of concatenation levels. Furthermore, a quantum circuit containing A gates can be simulated with probability of error at most ϵ with a spacetime overhead which scales as $O[\text{poly}(\log A/\epsilon)A]$ [28].

In this study, fault-tolerant syndrome measurement and error correction are implemented using Steane's method (see Sec. II for a detailed description of the method) in order to take advantage of the CSS structure of the codes. Error correction steps are interleaved between the implementation of each fault-tolerant gate. Following Ref. [18], at the first level of concatenation, each logical gate is represented by a 1-exrec as illustrated in Fig. 1. Hence, the components in a level-1 logical gate will consist of state preparation and measurement, physical gates and memory locations. We use a recursive simulation for higher levels of concatenation where a fault-tolerant gate at level k is constructed by replacing each level-0 location in the level-$(k-1)$ logical gate by the corresponding level-1 rectangle.

The noise threshold for the 49- and 105-qubit codes is computed by considering a depolarizing noise model for each physical (level-0) location. The depolarizing channel for a single qubit is defined by the quantum operation

$$\varepsilon(\rho) = \left(1 - \frac{3p}{4}\right)\rho + \frac{p}{4}(X\rho X + Y\rho Y + Z\rho Z), \qquad (18)$$

where p will be referred to as the physical error rate. Using the same parameters as Eq. (18), the depolarizing channel is generalized to all forms of quantum operations as follows:

(1) A noisy CNOT gate is modeled as applying a CNOT gate followed by, with probability $\frac{15p}{16}$, a two-qubit Pauli error drawn uniformly and independently from $\{I,X,Y,Z\}^{\otimes 2} \setminus \{I \otimes I\}$.

(2) A noisy preparation of the $|0\rangle$ state is modeled as the ideal preparation of the $|0\rangle$ state with probability $1 - \frac{p}{2}$ and $|1\rangle = X|0\rangle$ with probability $\frac{p}{2}$ (we use $\frac{p}{2}$ instead of $\frac{p}{4}$ since Y errors have the same effect as X errors). Similarly, the noisy preparation of the $|+\rangle$ state is modeled as the ideal preparation of the $|+\rangle$ state with probability $1 - \frac{p}{2}$ and $|-\rangle = Z|+\rangle$ with probability $\frac{p}{2}$.

(3) A noisy measurement in the Z basis is modeled by applying a Pauli X error with probability $\frac{p}{2}$ followed by an ideal measurement in the Z basis. Similarly, a noisy measurement in the X basis is modeled by applying a Pauli Z error with probability $\frac{p}{2}$ followed by an ideal measurement in the X basis.

(4) A single-qubit gate error or storage error is modeled by applying the ideal gate (identity gate for a resting qubit) with probability $1 - \frac{3p}{4}$. With probability $\frac{3p}{4}$, the ideal gate is implemented followed by a Pauli error chosen uniformly from the set $\{X,Y,Z\}$.

To determine the probability of having a logical fault at the output of an exrec, we first define the notion of a malignant error event. Let $|\psi_1\rangle$ be a single- or two-qubit logical state obtained by applying ideal decoders immediately after the LEC circuit and $|\psi_2\rangle$ the logical state obtained by applying ideal decoders immediately after the TEC. The event mal_E is defined as $|\psi_2\rangle = EU|\psi_1\rangle$ where E is a single- or two-qubit error and U is the desired gate. We now describe our simulation protocol to obtain estimates of the event mal_E for various logical gates:

(1) Given a 1-exrec encoding a particular gate, we fix a particular value of p and N, where N corresponds to the total number of iterations that the depolarizing channel is applied to the 1-exrec.

(2) We would like to calculate the probability of the event mal_E *conditioned* on acceptance of all ancilla states in the LEC and TEC circuits. For every location in the ancilla state preparation and verification circuits, we insert Pauli errors according to the depolarizing error model described above. We propagate the errors through the encoding and verification circuits of $|\bar{0}\rangle$ and $|\overline{+}\rangle$. If no errors are detected at the ancilla measurement locations of the ECs, we record the errors that lead to acceptance in the matrices $M_{|\bar{0}\rangle}$ and $M_{|\overline{+}\rangle}$. Each row of $M_{|\bar{0}\rangle,|\overline{+}\rangle}$ will correspond to an error of the form $e = [k,i,l,t]$ where k corresponds to the error type, i and l encode the logical and physical qubit numbers, and t is the particular time step where the error occurred. Note that since an ensemble of errors can combine leading to acceptance of the ancillas, the rows of $M_{|\bar{0}\rangle,|\overline{+}\rangle}$ will be grouped into several blocks. We repeat this process until the number of blocks of $M_{|\bar{0}\rangle,|\overline{+}\rangle}$ reaches a predefined size (in most cases, we chose the size to be 10^6 as we believe it to be an accurate representation of the noise for the state preparation circuits).

(3) For all the remaining locations of the 1-exrec (excluding the state preparation encoding and verification circuits), we insert errors according to the depolarizing noise model. Furthermore, we randomly pick a block from the matrices $M_{|\bar{0}\rangle}$ and $M_{|\overline{+}\rangle}$. The combined errors propagate through the 1-exrec and we project the final output errors back onto the code space. Steps 1–3 are repeated N times.

(4) For single-qubit gates, the logical errors are recorded into the vector $v_1 = [a_X,a_Y,a_Z]$ where, for example, a_X corresponds to the number of logical X errors that occurred after N iterations. For a two-qubit gate, the errors are recorded into a vector with 15 columns, one for each error type. For a gate G at a physical error rate p, the estimate of the probability of the event mal_E is given by $\Pr[\text{mal}_E|G,p] = a_E/N$. Hence, larger values of N will lead to better estimates of $\Pr[\text{mal}_E|G,p]$ by reducing the standard deviation.

For a 1-exrec encoding a logical gate G, the pseudothreshold is defined as the crossing point $p = p_G^1(p)$, where $p_G^1(p) = \sum_{E_i} \Pr[\text{mal}_{E_i}|G,p]$ for all possible logical errors E_i for a given logical gate G. The pseudothreshold thus corresponds to the physical error rate below which the logical error rate is smaller than the physical error rate. To obtain the asymptotic threshold, we first upper bound $\Pr[\text{mal}_E^{(1)}|G,p]$ (the probability

TABLE III. Lower bounds for the pseudothreshold and asymptotic threshold results for the Hadamard, T gate, and CNOT gate of the 105-qubit code. The Hadamard asymptotic threshold is larger than its pseudothreshold resulting from the double protection of the CNOT gates as seen by the high CNOT pseudothreshold.

	Pseudothreshold	Asymptotic threshold
CNOT gate	$(2.11 \pm 0.02)\times 10^{-3}$	$(1.95 \pm 0.01)\times 10^{-3}$
T gate	$(4.89 \pm 0.11)\times 10^{-4}$	$(1.58 \pm 0.02)\times 10^{-3}$
Hadamard gate	$(4.47 \pm 0.29)\times 10^{-5}$	$(1.28 \pm 0.02)\times 10^{-3}$
105-qubit	$(4.47 \pm 0.29)\times 10^{-5}$	$(1.28 \pm 0.02)\times 10^{-3}$

of a malignant event mal_E at the first level of concatenation) by

$$\Pr\left[\mathrm{mal}_E^{(1)}|G,p\right] \leqslant \sum_{k=\lceil \frac{d^*}{2} \rceil}^{L_G} c(k)p^k \equiv \Gamma_{G,E}^{(1)}, \qquad (19)$$

where the coefficients $c(k)$ are positive integers that parametrize the number of possible weight-k errors that can lead to a logical fault, L_G is the total number of circuit locations in the logical gate G, and d^* characterizes the minimal distance of a given logical gate. As was shown in Ref. [19], $\Gamma_{G,E}^{(1)}(p)$ is a polynomial that is monotonically increasing as a function of the physical error rate and serves as an upper bound for the failure probability at the first level of concatenation. Following Refs. [19,21], to obtain the probability of having the event mal_E at the second level of concatenation, we can treat each level-1 exrec in the level-2 simulation as a physical location with a modified noise model (no longer depolarizing) given by the $\Gamma_{G,E}^{(1)}(p)$ terms. This procedure can be generalized to the kth level of concatenation, enabling the upper bound on $\Pr[\mathrm{mal}_E^{(k)}|G,p]$ to be given by

$$\Pr\left[\mathrm{mal}_E^{(k)}|G,p\right] \leqslant \sum_{l=\lceil \frac{d^*}{2} \rceil}^{L_G} c(l)\left(\Gamma_{G,E}^{(k-1)}\right)^l \equiv \Gamma_{G,E}^{(k)}, \qquad (20)$$

where the $c(l)$ coefficients are the same as those in Eq. (19). It was then showed in Ref. [21] that for physical error rates p smaller than the crossing point between $\Gamma_{G,E}^{(1)}$ and $\Gamma_G^{(2)}$ (which we define to be $p_{th,G}$), the logical error rates for the mth level of concatenation ($m \geqslant 2$) could be upper bounded by

$$\Pr\left[\mathrm{mal}_E^{(m)}|G,p\right] \leqslant \Gamma_{G,E}^{(m)} \leqslant \epsilon^{\lceil \frac{d^*}{2} \rceil^{m-2}+1}\Gamma_{G,E}^{(1)}. \qquad (21)$$

Due to the exponential suppression seen in Eq. (21) of the logical error rate for $p \leqslant p_{th,G}$, $p_{th,G}$, this serves as a lower bound for the asymptotic threshold of the logical gate G.

In Ref. [21], the pseudothreshold and asymptotic threshold for the 105-qubit code were calculated for the H, T, and CNOT gates. Note that $\Gamma_{G,E}^{(m)}$ was also calculated for all other location types (storage, measurement, and state preparation) and were shown to have much higher threshold than the logical gates. The results are summarized in Table III, and Fig. 2 illustrates the noise behavior of the Hadamard and CNOT gates for several concatenation levels. An important feature

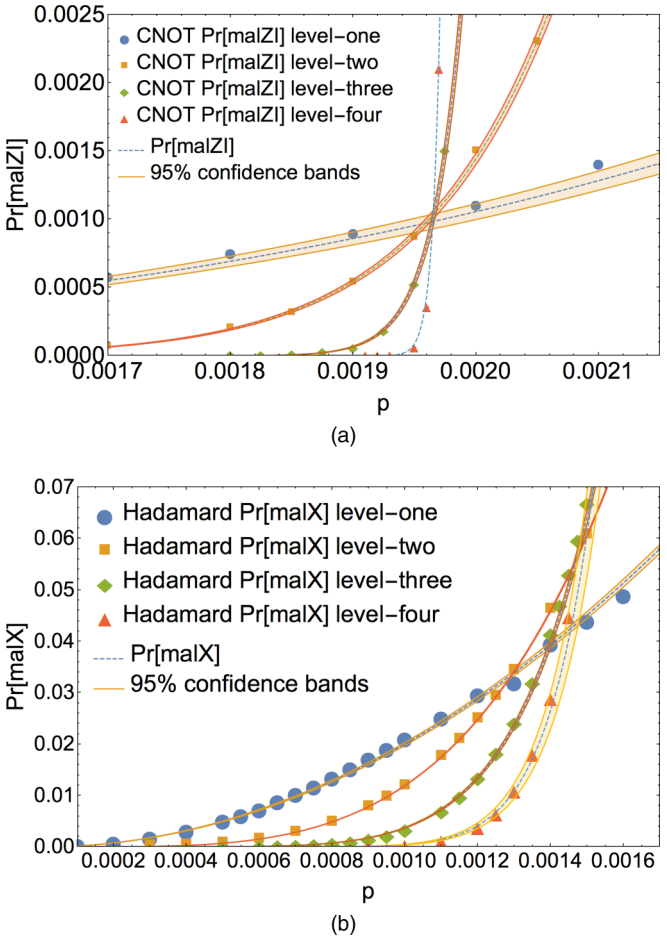

FIG. 2. Probability $\Pr[\mathrm{mal}_E]$ of logical error as function of physical error rate p for the level-1, level-2, and level-3 logical (a) CNOT and (b) Hadamard gates of the 105-qubit code. The crossing point between the level-1 and level-2 curves allows for the determination of a lower bound for the asymptotic threshold for each of the logical gates. The CNOT gate exhibits a much lower logical error rate than the Hadamard at the first level.

of these results is that, for the H and T gates, noise can be further suppressed by several orders of magnitude even for physical error rates above the pseudothreshold. To understand this type of noise behavior, it is important to point out that since the CNOT gate is transversal in both the 7- and 15-qubit codes, it receives a double protection from both codes. The double protection results in a much larger pseudothreshold for the CNOT gate compared to the pseudothresholds for the H and T gates. Furthermore, the asymptotic threshold of the CNOT is comparable to its pseudothreshold. Another important feature is that CNOT gates are the gates that are most present in the Steane's EC circuits as well as in the logical H and T gate circuits. Consequently, when going to higher levels of concatenation, and for error rates below the CNOT asymptotic threshold, all the CNOT gates will be less likely to fail compared to the previous level of concatenation. Even if the physical H and T gate locations are more likely to fail, the lower logical failure rates of the CNOT gates will compensate, resulting in overall further noise suppression.

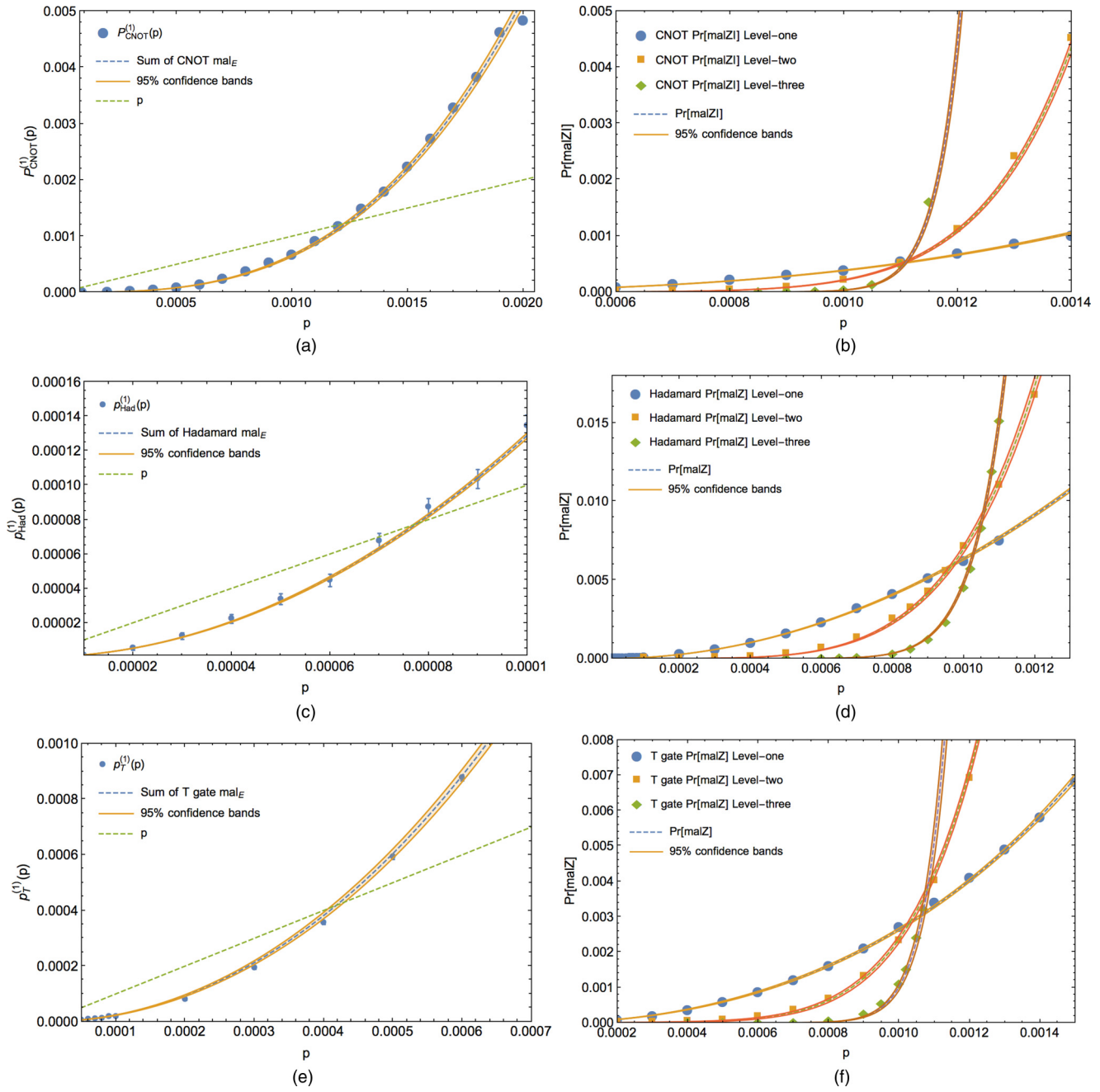

FIG. 3. All plots in Fig. 3 pertain to the 49-qubit code. The plots on the left column illustrate the probability of logical error as function of physical error rate for logical (a), (c) Hadamard and (e) T gates. The crossing point of the fitted curve allows for the determination of the level-1 pseudothreshold for each of the logical gates. The CNOT pseudothreshold is the largest among all three gates due to the double protection of the 7- and 15-qubit codes. The plots on the right column illustrate the polynomials upper bounding the probability of obtaining a logical error E for the first, second, and third levels of concatenation. The crossing point between the level-one and -two polynomials determines the asymptotic threshold for the gate under consideration. For the logical CNOT gate (b), it is the event mal$_{ZI}$ which limits the threshold value. The same behavior holds for the 105-qubit code. However, given that the effective distance of the 49-qubit code is 5 compared to 9 for the 105-qubit code (when the gate is transversal in both the 7- and 15-qubit codes), the 105-qubit code offers greater suppression for logical error rates below threshold. For the logical gate H (d) and T gate (f), mal$_Z$ limits the threshold value. Note that for the 105-qubit code, mal$_X$ limited the Hadamard threshold value. See Sec. V for more details.

We now provide the threshold results for the 49-qubit code. The plots in Fig. 3 illustrate $\Gamma^{(m)}_{G,E}$ for the Hadamard, T, and CNOT gates. As was the case for the 105-qubit code, the pseudothreshold of the 49-qubit code is limited by the Hadamard gate and is given by $p^{(1)}_{\text{Had}} = (7.76 \pm 0.17) \times 10^{-5}$. A lower bound on the asymptotic threshold was found to be $p_{th} = (9.69 \pm 0.28) \times 10^{-5}$. However, Fig. 3(c) shows a few noticeable differences with the 49-qubit code. First, logical Z errors are what limit the Hadamard asymptotic threshold for the 49-qubit code, not logical X errors as in the 105-qubit

TABLE IV. Lower bounds for the pseudothreshold and asymptotic threshold results for the Hadamard, T gate, and CNOT gate of the 49-qubit code. The Hadamard asymptotic threshold is larger than its pseudothreshold resulting from the double protection of the CNOT gates as seen by the high CNOT pseudothreshold.

	Pseudothreshold	Asymptotic threshold
CNOT gate	$(1.21 \pm 0.04)\times 10^{-3}$	$(1.10 \pm 0.01)\times 10^{-3}$
T gate	$(4.18 \pm 0.24)\times 10^{-4}$	$(1.03 \pm 0.03)\times 10^{-3}$
Hadamard gate	$(7.76 \pm 0.17)\times 10^{-5}$	$(9.69 \pm 0.28)\times 10^{-4}$
49-qubit	$(7.76 \pm 0.17)\times 10^{-5}$	$(9.69 \pm 0.28)\times 10^{-4}$

code. Second, the Hadamard circuit is less sensitive to input Z errors than in the 105-qubit code. Furthermore, the 49-qubit Hadamard pseudothreshold is larger than for the 105-qubit code. To understand these results, we first point out that the 49-qubit code Hadamard circuit contains four blocks consisting only of physical H and storage gates (no CNOT gates). As was explained in the Appendix, the cascading sequence of CNOT gates in the 15-qubit Hadamard circuit makes it very sensitive to input Z errors since any Z error that lands on the target of the CNOT gates (appearing before the physical H gate) will lead to a logical X error. Furthermore, Z errors occurring at CNOT or storage locations within the 15-qubit Hadamard circuit also play a dominant role in producing a logical X fault at the output of the circuit. The four blocks consisting only of physical H and storage gates will treat X and Z errors on the same footing and these blocks contain much fewer locations where Z errors can occur. A numerical simulation also showed that many of the errors leading to a logical Z fault were Z errors that occurred on CNOT gates *after* the physical H gate combined with Y errors on one of the single-qubit code blocks. We also note that for higher concatenation levels, given the smaller number of locations in a storage gate exrec, storage gates are less likely to fail than CNOT gates and so the single-qubit code blocks are less likely to acquire a logical fault. Consequently, the 49-qubit Hadamard circuit will produce less logical X errors compared to the circuit for the 105-qubit code. Since the 15-qubit code offers less protection against Z errors, logical Z errors become the dominant source of error for the Hadamard circuit.

Note that as in the 105-qubit code, the double protection of the CNOT gates from both the 7- and 15-qubit code results in a larger pseudothreshold and asymptotic threshold relative to the H and T gates. The threshold results for the 49-qubit code are summarized in Table IV. Although the 49-qubit code Hadamard pseudothreshold is larger than the 105-qubit code pseudothreshold, the asymptotic threshold is slightly smaller due to the lower CNOT pseudothreshold and asymptotic threshold.

VI. RESOURCE OVERHEAD FOR THE 49- AND 105-QUBIT CODES

A. Raw qubit overhead

In the simulation of a particular gate (say H or CNOT), we would like to obtain the resource overhead required to achieve a particular target logical error rate p_{target}. The overhead can be

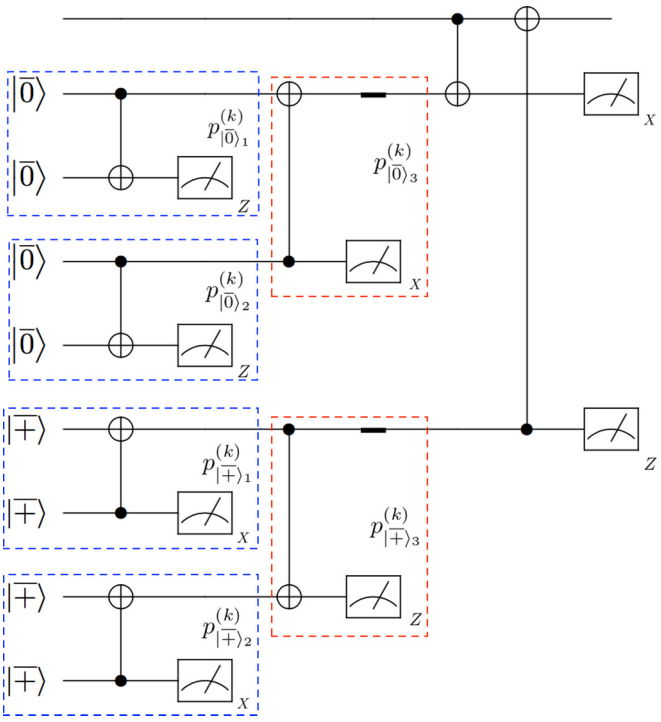

FIG. 4. Illustration of Steane's error correction circuit. The encoded $|\bar{0}\rangle$ and $|\bar{+}\rangle$ ancilla states are used to detect and correct Z and X errors. However, because the encoding circuits are not fault tolerant, extra $|\bar{0}\rangle$ and $|\bar{+}\rangle$ verifier states are used to detect errors during the encoding step. If an error is detected (or the -1 eigenvalue of a logical Pauli operator is measured), the ancilla states are rejected and the error correction round starts over. The terms $p^{(k)}_{|\bar{0}\rangle_i}$ and $p^{(k)}_{|\bar{+}\rangle_j}$ (where $i,j \in \{1,2\}$) correspond to the probabilities that no X (Z) errors are detected in the blocks $|\bar{0}\rangle_i$ ($|\bar{+}\rangle_j$). Conditioned on acceptance of the previous blocks, $p^{(k)}_{|\bar{0}\rangle_3}$ ($p^{(k)}_{|\bar{+}\rangle_3}$) denote the probabilities that no Z (X) errors are detected in the last verifier blocks. These probabilities will be used in the depolarizing noise overhead calculations of the 49- and 105-qubit codes.

measured in several ways. The raw qubit overhead measures how many physical qubits are required in the simulation of a logical gate to achieve a particular target logical error rate. The gate overhead measures the total number of gates used in the simulation of a logical gate in order to achieve a particular target logical error rate.

We begin with the analysis of the raw qubit overhead. Recall that Steane error correction (EC) is implemented by the circuit in Fig. 4 where the ancilla states $|\bar{0}\rangle$ and $|\bar{+}\rangle$ (for the 49- and 105-qubit codes) are prepared using the circuits of the Appendix.

We assume that the qubits used in the preparation of the ancilla states can be reused at each time step in the computation prior to a measurement in the X or Z basis. Before proceeding with the overhead calculation, we provide a few definitions. Let $|\bar{0}\rangle^{(k)}$ and $|\bar{+}\rangle^{(k)}$ correspond to the state-preparation circuits of $|0\rangle$ and $|+\rangle$ at the kth level of concatenation. In the first part of the Steane EC circuit, the ancilla states are verified for errors by entangling a pair of ancilla states of the same type and performing a measurement in the appropriate basis (see Appendix for more details). We define $n^{(k)}_{|\bar{0}\rangle}$ and $n^{(k)}_{|\bar{+}\rangle}$ to be the

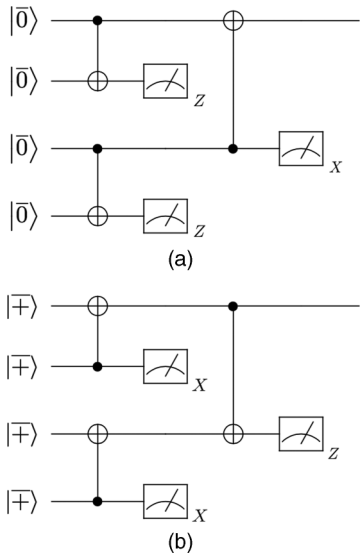

FIG. 5. (a) State preparation, including verification, for the ancilla state $|\bar{0}\rangle$. (b) State preparation circuit for the state $|\overline{+}\rangle$.

number of qubits required for the ancillas $|\bar{0}\rangle^{(k)}$ and $|\overline{+}\rangle^{(k)}$ to pass the verification test in the circuits of Figs. 5(a) and 5(b) at the kth level of concatenation. If an error is detected from the syndrome measurement, the ancilla is rejected and the process is repeated using a fresh batch of qubits which we assume are readily available. The probabilities of acceptance $p^{(k)}_{|\bar{0}\rangle_1}$ and $p^{(k)}_{|\overline{+}\rangle_1}$ were computed from a Monte Carlo algorithm. For an error correcting code with n physical qubits, we have at the first level of concatenation

$$n^{(1)}_{|\bar{0}\rangle} = \frac{2n\left(\frac{1}{p^{(1)}_{|\bar{0}\rangle_1}} + \frac{1}{p^{(1)}_{|\bar{0}\rangle_2}}\right)}{p^{(1)}_{|\bar{0}\rangle_3}}, \tag{22}$$

$$n^{(1)}_{|\overline{+}\rangle} = \frac{2n\left(\frac{1}{p^{(1)}_{|\overline{+}\rangle_1}} + \frac{1}{p^{(1)}_{|\overline{+}\rangle_2}}\right)}{p^{(1)}_{|\overline{+}\rangle_3}}. \tag{23}$$

Note that the term $2n/p^{(1)}_{|\bar{0}\rangle_j}$ (where $j \in \{1,2\}$) corresponds to the expected number of qubits for preparing a $|\bar{0}\rangle$ state free of X errors (the first step in the ancilla verification test). Hence, Eq. (22) corresponds to the expected number of qubits for the full $|\bar{0}\rangle$ ancilla verification test. The analogous equation holds for $|\overline{+}\rangle$ state preparation. Defining $n^{(k)}_{EC}$ to be the raw qubit overhead for a Steane EC circuit at level-k (excluding the overhead from the data qubits), we have that

$$n^{(1)}_{\mathrm{EC}} = n^{(1)}_{|\bar{0}\rangle} + n^{(1)}_{|\overline{+}\rangle}. \tag{24}$$

Since a logical CNOT gate consists of four EC circuits and two logical qubits, the level-1 overhead for the CNOT gate is

$$q^{(1)}_{\mathrm{CNOT}} = 2n^{(1)}_{\mathrm{EC}} + 2n. \tag{25}$$

Note the factor of 2 and not 4 in front of the n_{EC} term. This is due to the fact that we assume that the qubits used in the LEC circuit can be reused in the TEC circuits. Similarly, the level-1

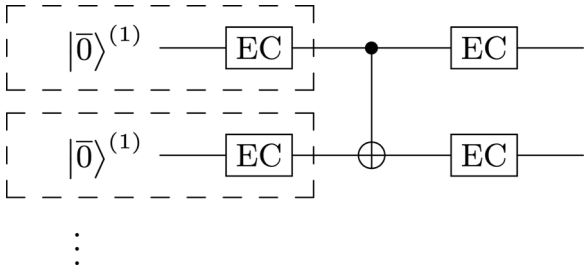

FIG. 6. Typical segment in a level-2 $|\bar{0}\rangle^{(2)}$ or $|\overline{+}\rangle^{(2)}$ illustrating the overlapping EC circuits between the $|\bar{0}\rangle^{(1)}$ (could also be $|\overline{+}\rangle^{(1)}$) and the following logical gate (a CNOT gate in this example). The overhead of the first EC circuit needs to be taken into account in the calculation of $n^{(2)}_{|\bar{0}\rangle}$ and $n^{(2)}_{|\overline{+}\rangle}$. Each dashed box has an overhead of $n^{(1)}_{\mathrm{EC}} + n$.

Hadamard qubit overhead is given by

$$q^{(1)}_{\mathrm{Had}} = n^{(1)}_{\mathrm{EC}} + n. \tag{26}$$

At the second level of concatenation, $|\bar{0}\rangle^{(2)}$ will contain $|\bar{0}\rangle^{(1)}$ and $|\overline{+}\rangle^{(1)}$ state-preparation circuits which will be followed by an EC circuit. However, all other gates (CNOTs and storage) will also contain their respective LEC and TEC circuits (since recall that for a level-2 simulation each physical gate is replaced by a level-1 exrec). The EC circuits of $|\bar{0}\rangle^{(1)}$ and $|\overline{+}\rangle^{(1)}$ will overlap with the LEC circuit of the gate that follows (see Fig. 6) and so it is important to take into account the overhead of the overlapping EC circuit. The full EC circuits (including the data qubits) each have an overhead of $n^{(1)}_{\mathrm{EC}} + n$. We assume that the qubits that were used in the EC circuit following $|\bar{0}\rangle^{(1)}$ and $|\overline{+}\rangle^{(1)}$ can be reused for all other EC circuits that follow. Hence, we only take into account the overhead of the *first* EC circuit. The entire $|\bar{0}\rangle^{(2)}$ circuit will have an overhead of $n(n^{(1)}_{\mathrm{EC}} + n)$. Generalizing to the kth concatenation level (for $k \geqslant 2$), we have the recursive relation

$$n^{(k)}_{|\bar{0}\rangle} = \frac{2n\left(n^{(k-1)}_{\mathrm{EC}} + n^{(k-1)}\right)\left(\frac{1}{p^{(k)}_{|\bar{0}\rangle_1}} + \frac{1}{p^{(k)}_{|\bar{0}\rangle_2}}\right)}{p^{(k)}_{|\bar{0}\rangle_3}}, \tag{27}$$

$$n^{(k)}_{|\overline{+}\rangle} = \frac{2n\left(n^{(k-1)}_{\mathrm{EC}} + n^{(k-1)}\right)\left(\frac{1}{p^{(k)}_{|\overline{+}\rangle_1}} + \frac{1}{p^{(k)}_{|\overline{+}\rangle_2}}\right)}{p^{(k)}_{|\overline{+}\rangle_3}}. \tag{28}$$

The EC circuit at level-k has an overhead given by

$$n^{(k)}_{\mathrm{EC}} = n^{(k)}_{|\bar{0}\rangle} + n^{(k)}_{|\overline{+}\rangle}. \tag{29}$$

The CNOT and Hadamard overheads at level-k are then given by

$$q^{(k)}_{\mathrm{CNOT}} = 2n^{(k)}_{\mathrm{EC}} + 2n^k, \tag{30}$$

$$q^{(k)}_{\mathrm{Had}} = n^{(k)}_{\mathrm{EC}} + n^k. \tag{31}$$

B. Gate overhead

In this section, we focus on the gate overhead for the simulation of a logical Hadamard and logical CNOT gate. We define $g^{(k)}_{|\bar{0}\rangle}$ and $g^{(k)}_{|\overline{+}\rangle}$ to be the number of gates in a $|\bar{0}\rangle$

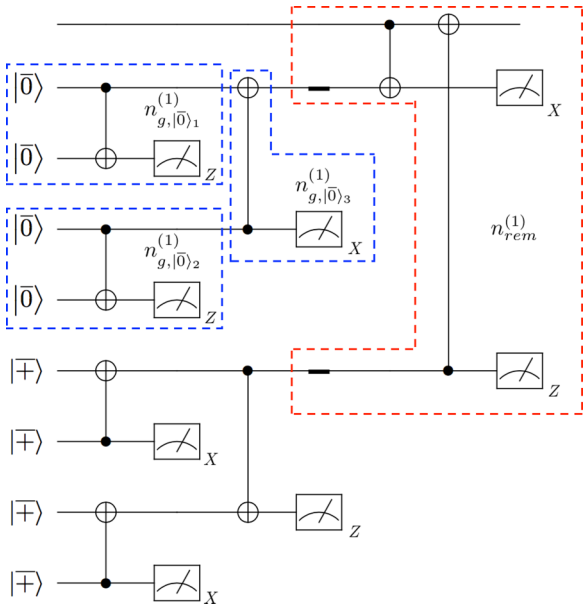

FIG. 7. Circuit illustrating the various definitions for the gate overhead calculation. $n^{(1)}_{g,|\overline{0}\rangle_1}$ corresponds to the gate overhead for the first part in the $|\overline{0}\rangle$ ancilla verification test at the first level of concatenation. Note that $n^{(1)}_{g,|\overline{0}\rangle_1} = n^{(1)}_{g,|\overline{0}\rangle_2}$. The term $n^{(1)}_{g,|\overline{0}\rangle_3}$ corresponds to the gate overhead for the final step in the $|\overline{0}\rangle$ ancilla verification test. The $n^{(1)}_{g,|\overline{+}\rangle_j}$ terms are defined in an analogous way. Lastly, $n^{(1)}_{\text{rem}}$ is the gate overhead for the remaining part of the Steane EC circuit.

and $|\overline{+}\rangle$ circuit at level-k. Since $|\overline{0}\rangle$ and $|\overline{+}\rangle$ circuits consist of physical $|0\rangle$ and $|+\rangle$ states as well as CNOT and storage gates, we can define α_j to be the number of locations of type j in a $|\overline{0}\rangle^{(1)}$ circuit and β_j to be the number of locations of type j in a $|\overline{+}\rangle^{(1)}$ circuit. In this case, we can write

$$g^{(1)}_{|\overline{0}\rangle} = \alpha_{\text{CNOT}} + \alpha_{\text{mem}} + \alpha_{|0\rangle} + \alpha_{|+\rangle}, \tag{32}$$

$$g^{(1)}_{|\overline{+}\rangle} = \beta_{\text{CNOT}} + \beta_{\text{mem}} + \beta_{|0\rangle} + \beta_{|+\rangle}. \tag{33}$$

To obtain the gate overhead of a Steane EC circuit at the first level of concatenation, we can divide the EC circuit into several components and compute the overhead for each component. We define $n^{(1)}_{g,|\overline{0}\rangle_j}$ to be the gate overhead for the $|\overline{0}\rangle$ ancilla verification components, $n^{(1)}_{g,|\overline{+}\rangle_j}$ to be the gate overhead for the $|\overline{+}\rangle$ ancilla verification components, and $n^{(1)}_{\text{rem}}$ to be the gate overhead for the remaining EC circuit (see Fig. 7 for a circuit description).

Since CNOT and measurement locations can be implemented transversally, they each contribute a factor of n to the gate overhead at the first level so that

$$n^{(1)}_{g,|\overline{0}\rangle_1} = n^{(1)}_{g,|\overline{0}\rangle_2} = 2\big(g^{(1)}_{|\overline{0}\rangle} + n\big), \tag{34}$$

$$n^{(1)}_{g,|\overline{+}\rangle_1} = n^{(1)}_{g,|\overline{+}\rangle_2} = 2\big(g^{(1)}_{|\overline{+}\rangle} + n\big), \tag{35}$$

$$n^{(1)}_{g,|\overline{0}\rangle_3} = n^{(1)}_{g,|\overline{+}\rangle_3} = 2n, \tag{36}$$

$$n^{(1)}_{\text{rem}} = 6n. \tag{37}$$

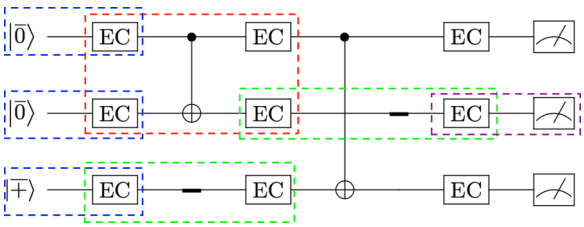

FIG. 8. Typical sequence of gates in a level-k exrec. The EC preceding the state-preparation circuit overlaps with the LEC of the following gate. The TEC of the last gate overlaps with the EC prior to a performing a measurement. In order to avoid overcounting gates appearing in overlapping ECs, we use the overhead of logical gates with truncated LECs in the counting procedure. The EC circuit arising from a measurement location will be included in the TEC of the previous gate.

Taking into account that the ancilla states are rejected if a nontrivial error is detected at the measurement locations, we can compute the expected number of gates for a $|\overline{0}\rangle$ and $|\overline{+}\rangle$ verification test (defined as $n^{(1)}_{g,\text{EC}|\overline{0}\rangle}$ and $n^{(1)}_{g,\text{EC}|\overline{+}\rangle}$) according to their success probabilities as

$$n^{(1)}_{g,\text{EC}|\overline{0}\rangle} = \frac{n^{(1)}_{g,|\overline{0}\rangle_1}\left(\frac{1}{p^{(1)}_{|\overline{0}\rangle_1}} + \frac{1}{p^{(1)}_{|\overline{0}\rangle_2}}\right) + n^{(1)}_{g,|\overline{0}\rangle_3}}{p^{(1)}_{|\overline{0}\rangle_3}}, \tag{38}$$

$$n^{(1)}_{g,\text{EC}|\overline{+}\rangle} = \frac{n^{(1)}_{g,|\overline{+}\rangle_1}\left(\frac{1}{p^{(1)}_{|\overline{+}\rangle_1}} + \frac{1}{p^{(1)}_{|\overline{+}\rangle_2}}\right) + n^{(1)}_{g,|\overline{+}\rangle_3}}{p^{(1)}_{|\overline{+}\rangle_3}}. \tag{39}$$

From Eqs. (37) and (39), the total gate overhead $n^{(1)}_{g,\text{EC}}$ for an EC circuit at the first level of concatenation is given by

$$n^{(1)}_{g,\text{EC}} = n^{(1)}_{g,\text{EC}|\overline{0}\rangle} + n^{(1)}_{g,\text{EC}|\overline{+}\rangle} + n^{(1)}_{\text{rem}}. \tag{40}$$

Using Eq. (40), the overhead of a level-1 CNOT and storage exrec is given by

$$g^{(1)}_{\text{CNOT}} = 4n^{(1)}_{g,\text{EC}} + n, \tag{41}$$

$$g^{(1)}_{\text{mem}} = 2n^{(1)}_{g,\text{EC}} + n. \tag{42}$$

The logical Hadamard circuit consists of storage, CNOT, and physical Hadamard gates. Defining γ_j to be the number of gates of type j in the level-1 logical Hadamard circuit and using Eq. (42), the gate overhead for the level-1 Hadamard exrec is

$$g^{(1)}_{\text{Had}} = 2n^{(1)}_{g,\text{EC}} + \gamma_{\text{CNOT}} + \gamma_{\text{mem}} + \gamma_{\text{Had}}. \tag{43}$$

Before obtaining the overhead at higher levels of concatenation, it is important to point out that consecutive exrec's will have overlapping ECs. In a recursive calculation, if we were to use the overhead terms $g^{(k-1)}_j$ at the kth concatenation level in the counting procedure, we would be overcounting the gates appearing in overlapping ECs (see Fig. 8). The overcounting can be avoided by ignoring the gate overhead in LECs from the $(k-1)$-exrec's appearing in a k-exrec. For a t-qubit gate, we define

$$\tilde{g}^{(k)}_j = g^{(k)}_j - tn^{(k)}_{g,\text{EC}}. \tag{44}$$

Therefore, in the overhead counting of a k-exrec, we will include the contributions from $\tilde{g}_j^{(k-1)}$ terms instead of $g_j^{(k-1)}$.

We now consider the gate overhead at the kth concatenation for $k \geqslant 2$. The gate overhead in a $|\overline{0}\rangle$ and $|\overline{+}\rangle$ circuit at level-k is given by

$$g_{|\overline{0}\rangle}^{(k)} = \alpha_{\text{CNOT}}\tilde{g}_{\text{CNOT}}^{(k-1)} + \alpha_{\text{mem}}\tilde{g}_{\text{mem}}^{(k-1)} + \alpha_{|0\rangle}g_{|\overline{0}\rangle}^{(k-1)} + \alpha_{|+\rangle}g_{|\overline{+}\rangle}^{(k-1)} + n_{g,\text{EC}}^{(k-1)}, \tag{45}$$

$$g_{|\overline{+}\rangle}^{(k)} = \beta_{\text{CNOT}}\tilde{g}_{\text{CNOT}}^{(k-1)} + \beta_{\text{mem}}\tilde{g}_{\text{mem}}^{(k-1)} + \beta_{|0\rangle}g_{|\overline{0}\rangle}^{(k-1)} + \beta_{|+\rangle}g_{|\overline{+}\rangle}^{(k-1)} + n_{g,\text{EC}}^{(k-1)}. \tag{46}$$

Note that the $n_{g,\text{EC}}^{(k-1)}$ term in Eq. (46) was included to take into account the gate overhead from the level-$(k-1)$ EC circuit that precedes $|\overline{0}\rangle^{(k)}$ and $|\overline{+}\rangle^{(k)}$.

In the first part of the $|\overline{0}\rangle$ ancilla verification test, we must include the contributions from the two $|\overline{0}\rangle^{(k)}$ circuits, the CNOT exrec, and the measurement locations. Since the EC circuit prior to performing a measurement is included in the contribution from the TEC of the CNOT gate, level-k measurement locations will always have a gate overhead of n^k so that

$$n_{g,|\overline{0}\rangle_1}^{(k)} = n_{g,|\overline{0}\rangle_2}^{(k)} = 2g_{|\overline{0}\rangle}^{(k)} + n\tilde{g}_{\text{CNOT}}^{(k-1)} + n^k, \tag{47}$$

$$n_{g,|\overline{+}\rangle_1}^{(k)} = n_{g,|\overline{+}\rangle_2}^{(k)} = 2g_{|\overline{+}\rangle}^{(k)} + n\tilde{g}_{\text{CNOT}}^{(k-1)} + n^k, \tag{48}$$

$$n_{g,|\overline{0}\rangle_3}^{(k)} = n_{g,|\overline{+}\rangle_3}^{(k)} = n\tilde{g}_{\text{CNOT}}^{(k-1)} + n^k, \tag{49}$$

$$n_{\text{rem}}^{(k)} = 2n\left(\tilde{g}_{\text{CNOT}}^{(k-1)} + \tilde{g}_{\text{mem}}^{(k-1)}\right) + 2n^k. \tag{50}$$

The overhead for the full ancilla verification procedure is obtained in the same way as in Eq. (39):

$$n_{g,\text{EC}|\overline{0}\rangle}^{(k)} = \frac{n_{g,|\overline{0}\rangle_1}^{(k)}\left(\frac{1}{p_{|\overline{0}\rangle_1}^{(k)}} + \frac{1}{p_{|\overline{0}\rangle_2}^{(k)}}\right) + n_{g,|\overline{0}\rangle_3}^{(k)}}{p_{|\overline{0}\rangle_3}^{(k)}}, \tag{51}$$

$$n_{g,\text{EC}|\overline{+}\rangle}^{(k)} = \frac{n_{g,|\overline{+}\rangle_1}^{(k)}\left(\frac{1}{p_{|\overline{+}\rangle_1}^{(k)}} + \frac{1}{p_{|\overline{+}\rangle_2}^{(k)}}\right) + n_{g,|\overline{+}\rangle_3}^{(k)}}{p_{|\overline{+}\rangle_3}^{(k)}}. \tag{52}$$

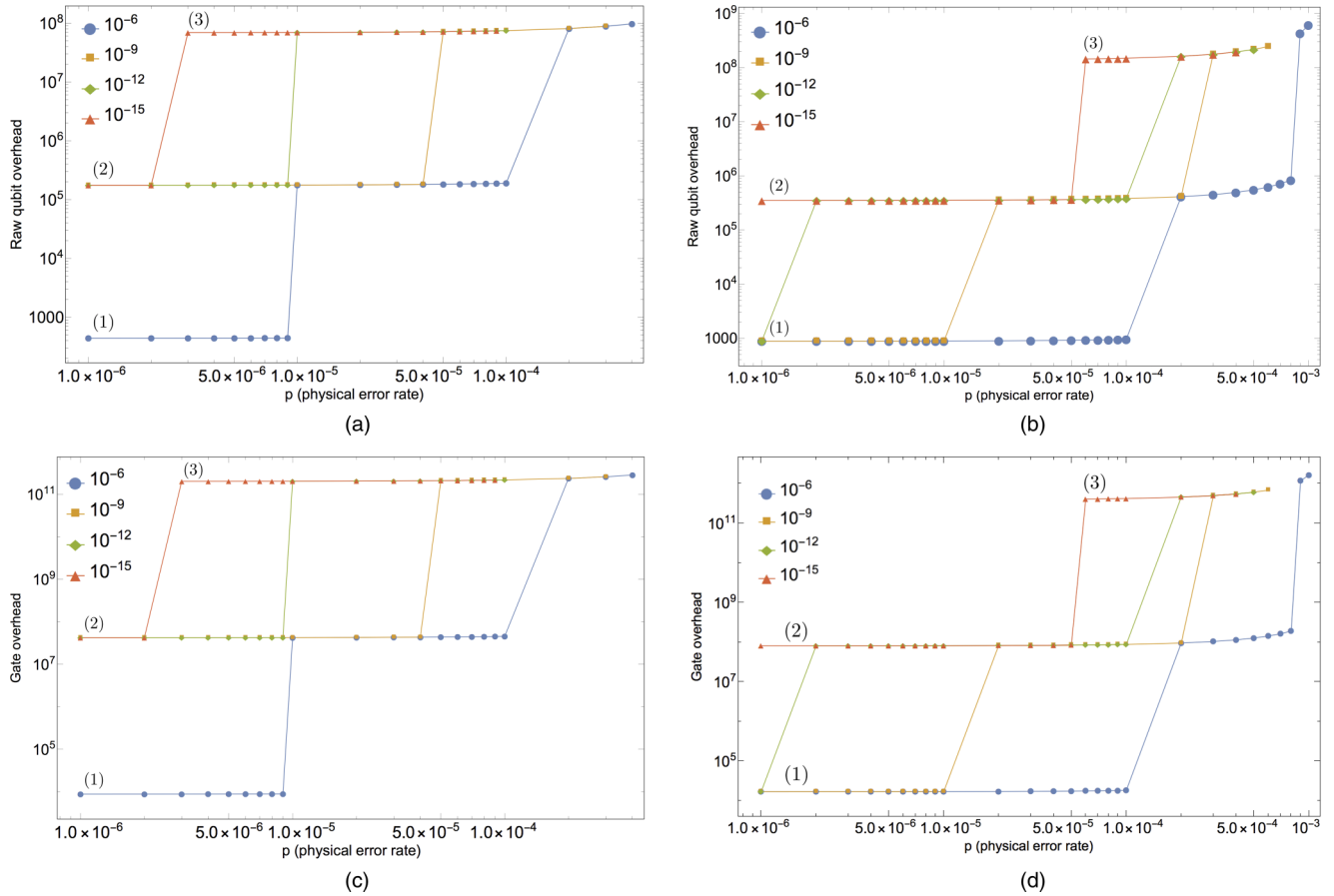

FIG. 9. Physical qubit and gate overhead log - log plots for the 49-qubit code. The plots in (a) and (c) illustrate the overhead for the Hadamard gate while the plots in (b) and (d) illustrate the overhead for the CNOT gate. The numbers in parentheses indicate the level of concatenation required to achieve the target error rates shown in the legend. The plots have a steplike function behavior since above certain physical error rates, a higher level of concatenation is required to achieve the particular target error rate. For $p_{\text{target}} = 10^{-15}$ and a physical error rate $p = 10^{-5}$, roughly 10^8 physical qubits are required to encode one logical Hadamard gate.

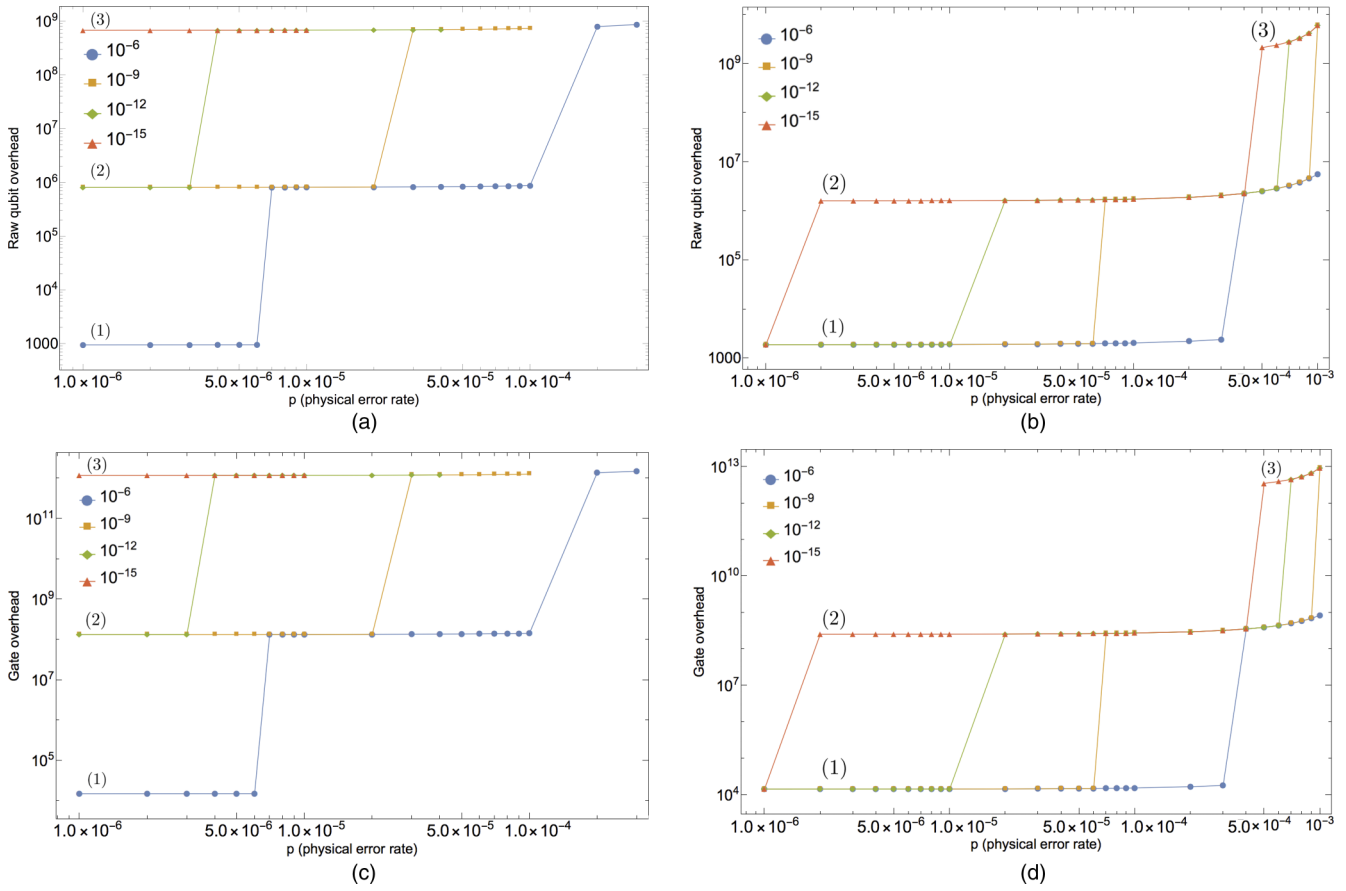

FIG. 10. Physical qubit and gate overhead log - log plots for the 105-qubit code. The plots in (a) and (c) illustrate the overhead for the Hadamard gate while the plots in (b) and (d) illustrate the overhead for the CNOT gate. The numbers in parentheses indicate the level of concatenation required to achieve the target error rates shown in the legend. The plots have a steplike function behavior since above certain physical error rates, a higher level of concatenation is required to achieve the particular target error rate. For $p_{\text{target}} = 10^{-15}$ and a physical error rate $p = 10^{-5}$, roughly 10^9 physical qubits are required to encode one logical Hadamard gate.

The overhead for the complete EC circuit at level-k is then the sum of the contributions from $n^{(k)}_{g,\text{EC}|\bar{0}\rangle}$, $n^{(k)}_{g,\text{EC}|\bar{+}\rangle}$, and $n^{(k)}_{\text{rem}}$,

$$n^{(k)}_{g,\text{EC}} = n^{(k)}_{g,\text{EC}|\bar{0}\rangle} + n^{(k)}_{g,\text{EC}|\bar{+}\rangle} + n^{(k)}_{\text{rem}}. \tag{53}$$

It is then straightforward to obtain the overhead for the CNOT, storage, and Hadamard exrecs, which we state below:

$$g^{(k)}_{\text{CNOT}} = 4n^{(k)}_{g,\text{EC}} + n\tilde{g}^{(k-1)}_{\text{CNOT}}, \tag{54}$$

$$g^{(k)}_{\text{mem}} = 2n^{(k)}_{g,\text{EC}} + n\tilde{g}^{(k-1)}_{\text{mem}}, \tag{55}$$

$$g^{(k)}_{\text{Had}} = 2n^{(k)}_{g,\text{EC}} + \gamma_{\text{CNOT}}\tilde{g}^{(k-1)}_{\text{CNOT}} + \gamma_{\text{mem}}\tilde{g}^{(k-1)}_{\text{mem}} + \gamma_{\text{Had}}\tilde{g}^{(k-1)}_{\text{Had}}. \tag{56}$$

C. 49- and 105-qubit code overhead results

In this section, we use the formalism of Secs. VI A and VI B to obtain the raw qubit and gate overhead results of the 49- and 105-qubit codes. Since the Hadamard gate limits the threshold value of both codes, we will focus on the overhead for performing a logical Hadamard and CNOT gate. Given a target logical error rate p_{target}, we can use the threshold results of Sec. V to determine the appropriate level of concatenation to reach p_{target}. We can then calculate the raw qubit overhead and gate overhead from Eqs. (30) and (31) and (54)–(56), respectively. The coefficients α_j, β_j, and γ_j are given in the Appendix.

Figures 9 and 10 illustrate the physical qubit and gate overhead for the 49- and 105-qubit codes. The key results are summarized in Table V. In the following discussion, we focus on a target logical error rate of $p_{\text{target}} = 10^{-15}$. For the 49-qubit logical Hadamard gate, if we limit the implementation to two levels of concatenation, it would require a level of precision of 2.36×10^{-6}, far below the asymptotic threshold, in order to reduce the logical error rate to 10^{-15}. The raw qubit and gate overheads are given by $(1.75 \pm 0.38) \times 10^5$ and $(4.20 \pm 0.92) \times 10^7$ in such a scenario. Going to a third level of concatenation allows for higher physical error rates, up to 8.47×10^{-5}, as the increase concatenation level will further reduce logical error rate. However, there is a tradeoff to increasing the concatenation level as it requires further resources. The raw qubit and gate overheads for an error rate of 8.47×10^{-5} are given by $(7.33 \pm 1.57) \times 10^7$ and $(2.14 \pm 0.66) \times 10^{11}$ (an increase of roughly three orders of magnitude). One aspect that is quite apparent in the overhead results is how well the 49-qubit logical Hadamard performs compared to the 105-qubit version. For the 105-qubit code, the

TABLE V. Overhead results for the 49- and 105-qubit codes. The first column indicates the code and corresponding logical gate for which the overhead is computed. The third column indicates the largest physical error rate p that can be achieved for the particular concatenation level so that the logical error rate is below $p_{\text{target}} = 10^{-15}$. The fourth and fifth columns give the qubit and gate overheads for the given physical error rate.

Code/Gate	Concatenation level	Physical error rate	Qubit overhead	Gate overhead
49-qubit Hadamard	1	N/A	N/A	N/A
	2	2.36×10^{-6}	$(1.75\pm0.38)\times10^{5}$	$(4.20\pm0.92)\times10^{7}$
	3	8.47×10^{-5}	$(7.33\pm1.57)\times10^{7}$	$(2.14\pm0.66)\times10^{11}$
49-qubit CNOT	1	N/A	N/A	N/A
	2	5.24×10^{-5}	$(3.60\pm0.79)\times10^{5}$	$(8.38\pm0.18)\times10^{7}$
	3	3.92×10^{-4}	$(1.94\pm0.37)\times10^{8}$	$(5.45\pm1.50)\times10^{11}$
105-qubit Hadamard	1	N/A	N/A	N/A
	2	5.35×10^{-7}	$(1.14\pm0.22)\times10^{6}$	$(1.86\pm0.34)\times10^{8}$
	3	1.60×10^{-5}	$(3.00\pm0.47)\times10^{9}$	$(4.82\pm0.73)\times10^{12}$
105-qubit CNOT	1	N/A	N/A	N/A
	2	4.56×10^{-4}	$(2.27\pm0.44)\times10^{6}$	$(3.55\pm0.66)\times10^{8}$
	3	1.39×10^{-3}	$(6.01\pm0.94)\times10^{9}$	$(9.28\pm1.40)\times10^{12}$

logical Hadamard gate can achieve the desired target error rate at the second concatenation level for physical error rates below 5.35×10^{-7}, as opposed to 2.36×10^{-6}, due to the complexity of the 105-qubit Hadamard construction compared to that of the 49-qubit code. The third level of concatenation pushes this number to 1.60×10^{-5}. Furthermore, for each concatenation level, the overheads are roughly an order of magnitude larger than for the 49-qubit logical Hadamard.

D. Comparison to the surface code overhead using state distillation

In this section, we aim to estimate the qubit (space) overhead for the implementation of fault-tolerant logic in the surface code. The primary obstacle to surface code implementations is the need to distill a special ancillary state for the purposes of implementing the T gate through gate teleportation. High-fidelity logical state preparation of this special state is obtained through a process called state distillation. State distillation is implemented using only logical Clifford gates, resulting in a nonstabilizer state that can be used to implement the T gate, called a magic state [31].

The idea behind magic state distillation begins by assuming the initial magic state has an error rate p. Then, by using multiple noisy logical magic states and near-perfect Clifford operations (since the Clifford operations are performed using fault-tolerant logical operations in the surface code), a higher-fidelity magic state can be distilled from multiple noisy states. Depending on the distillation scheme used, the output state of the scheme will have a fidelity of cp^b, for some constant value of b and c. The process can then be repeated with multiple copies of the newly distilled logical states to obtain a state with error rate $c(cp^b)^b = c^{b+1}p^{b^2}$. Iterating this process k times, the final logical output state will have an error rate

$$p_k = \frac{1}{c^{\frac{1}{b-1}}}(c^{\frac{1}{b-1}}p)^{b^k}. \tag{57}$$

The magic state distillation process can be probabilistic, yielding a distilled state based on the result of a set of measurements, and in general will succeed with probability $1/r$ (where r depends on the particular distillation scheme being used). Therefore, if n logical qubits are required for the distillation, and the probability of success for a given round is $1/r$, the total number of logical qubits required for k distillation rounds is given by $(rn)^k$. The number of qubits required is thus exponential in the number of rounds, however, the scheme remains theoretically efficient as the logical error rate is suppressed double exponentially. Table VI summarizes the different parameters for the magic state distillation schemes.

Suppose we would like to obtain a lower bound on the number of physical qubits that are required to implement the logical T gate in the surface code. Given a target error rate p_{target}, and a depolarizing physical error rate of p, we must first determine the number of distillation levels required to obtain the desired target rate, that is choose a value of k from Eq. (57) such that $p_k < p_{\text{target}}$. Having determined the level of distillation, recall that we argued that the Clifford gates must have low levels of noise with respect to the target error rate. Therefore, for a given distillation level, we will need to choose logical gates that have a small enough logical error rate. For the surface code, by considering the probability of a logical string being created given that the syndrome extraction scheme is eight time steps long, the logical gate error rate can be approximated as [17]

$$p_L(d) = d\binom{d}{\lceil d/2\rceil}(8p)^{\lceil d/2\rceil}, \tag{58}$$

where d is the distance of the code. Therefore, if the output of a given distillation level is given by p_k, as argued in Ref. [17], the

TABLE VI. Parameters for different magic state distillation schemes. The $|T\rangle$ and $|H\rangle$ schemes are given in Ref. [31], while the 10-to-2 qubit scheme is presented in Ref. [32].

Distillation type	Qubits	Error rate	Success prob.
$\|T\rangle$ type	5	$\frac{1}{5}(5p)^{2^k}$	1/6
$\|H\rangle$ type	15	$\frac{1}{\sqrt{35}}(\sqrt{35}p)^{3^k}$	$\approx 1 - p/15$
10-to-2	5	$\frac{1}{9}(9p)^{2^k}$	≈ 1

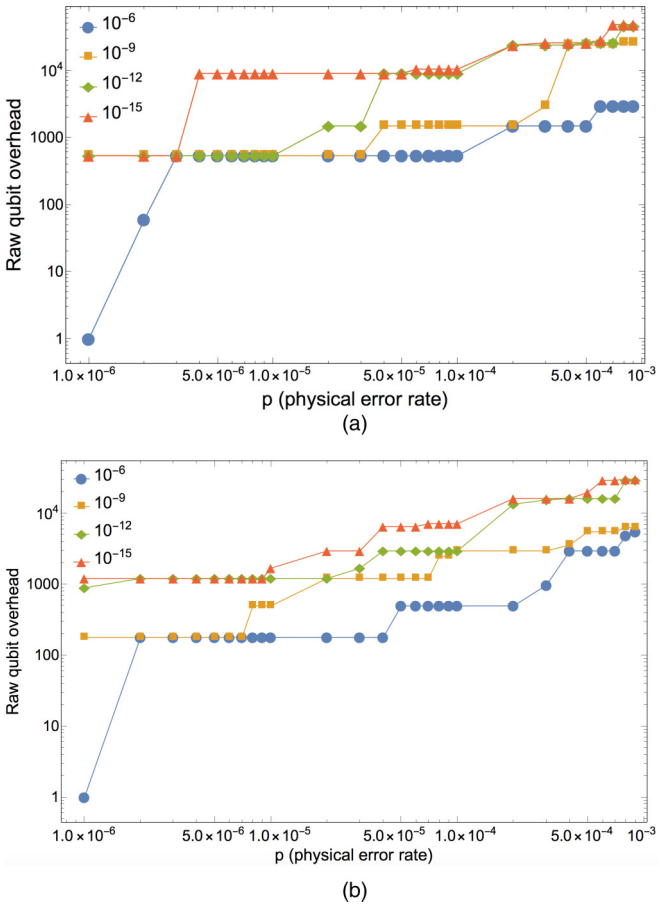

FIG. 11. (a) Physical qubit overhead for the distillation of the H-type magic state to be used in the implementation of the T gate for the surface code with different target gate error rates. (b) Distillation overhead for the Meier-Eastin 10-to-2 distillation scheme for the surface code. As can be seen in the above plots, for physical error rates of $p = 10^{-5}$, the Meier-Eastin 10-to-2 distillation scheme requires roughly 1000 physical qubits to implement the T gate. This is about an order of magnitude less than the physical qubits required for the distillation of the H-type magic state.

logical error rate must be small enough such that the resulting accumulation of errors from the distillation circuit does not negatively affect the distilled qubit, that is,

$$8 \times 1.25 \times n_q \times d \times p_L(d) \leqslant p_k, \tag{59}$$

where n_q is the number of qubits in the distillation scheme. Therefore, the distance must be chosen large enough to reduce the logical error rate to sufficiently small levels (assuming we are always below threshold). Having distilled at a given level, the logical qubits can be enlarged through a fault-tolerant growing operation, in order to be of the required distance for the next distillation level. In our lower bound of the number of qubits required, we will assume that this operation is done error free; in reality, this may slightly increase the number of physical qubits required. Therefore, given the required distance at each level, and the fact that the number of physical qubits for a distance d surface code is $(2d)^2$ (including the syndrome qubits), a lower bound of the distillation overhead can be obtained. Figure 11 illustrates the physical qubit overhead for two different distillation schemes used in the implementation of the T gate for the surface code. It can be seen that for a target logical error rate of $p_{\text{target}} = 10^{-15}$, both schemes yield an overhead on the order of 10^4 physical qubits for input error rates $10^{-4} < p < 10^{-3}$:

VII. CONCLUSION

In this paper, we reviewed the fault-tolerant construction of the 49- and 105-qubit codes obtained from concatenating Steane's 7-qubit code with the 15-qubit Reed-Muller code. One advantage of the concatenation scheme is that universal fault tolerance can be achieved without using state distillation protocols. Taking advantage of the CSS structure of the 7- and 15-qubit codes, the error correction blocks were constructed in the framework of Steane error correction. In the study of the performance of the concatenated codes, we obtained the threshold for the 105-qubit code for an adversarial noise model using malignant set counting and found it to be $(8.33 \pm 0.28) \times 10^{-6}$, which is slightly below that of the concatenated 7-qubit Steane code assisted by magic state distillation [18]. In order to obtain threshold estimates that would more accurately represent the noise seen in actual experiments, we computed the threshold value of the 49-qubit code for depolarizing noise using the techniques developed in Refs. [19,21]. The 49-qubit code depolarizing threshold value was found to be $(9.69 \pm 0.28) \times 10^{-4}$ which is competitive with the 105-qubit code threshold of $(1.28 \pm 0.02) \times 10^{-3}$ [21]. As was the case for the 105-qubit code, the threshold for the 49-qubit code was limited by the logical Hadamard gate.

We proceeded by developing general methods to compute the overhead of concatenated codes using Steane error correction and applied our methods to compute the physical qubit and gate overhead of the 49- and 105-qubit codes. We also computed the physical qubit overhead for surface codes implementing the T gate using the H-type magic state and the 10-to-2 distillation schemes [31,32]. Comparing the plots of Fig. 11 with those of Figs. 9 and 10, it is clear that surface code requires a smaller overhead than the 49- and 105-qubit codes given the sampled input error rates. For example, for an input error rate of $p = 5 \times 10^{-5}$, the 49-qubit code requires more than 10^7 physical qubits to implement a logical Hadamard gate compared to roughly 10^4 physical qubits to implement the T gate in the surface code.

To explain the differences in overhead, it is first important to point out that the surface code thresholds are about an order of magnitude larger than the studied concatenated code thresholds. Hence, for comparable input error rates below threshold, the logical noise rate would be further suppressed for the surface codes requiring the use of fewer qubits to achieve a particular target logical error rate, even when using more rounds of distillation. Furthermore, the size of the EC blocks for Steane error correction represents a big drawback for concatenated codes. To illustrate this, it can be seen in Fig. 10(a) that the second level of concatenation requires the use of more than 10^5 physical qubits for the 49-qubit code when implementing a logical Hadamard gate. Since the data qubits require the use $49^2 = 2401$ physical qubits, more than 97% of the overhead comes from the size of the EC blocks. Consequently, even if the thresholds for the

49- and 105-qubit codes were significantly improved, two levels of concatenation would require more than 10^5 physical qubits, which is more than the largest number of physical qubits used in the surface code implementation of the T gate for all sample error rates. The latter shows that in order to improve the overhead results of the studied concatenated codes, smaller EC blocks would need to be used that maintain the fault-tolerant properties of the concatenated scheme, even at the cost of lowering the asymptotic threshold. Additionally, it would be interesting to determine the resource overhead for the generalized construction of the concatenated model, using higher distance 2D and 3D color codes as the base codes for the implementation of the fault-tolerant universal gate set.

Finally, in this work we explore an alternative to magic state distillation in the form of code concatenation, however, other alternative schemes also exist [33–38]. It remains unclear whether such schemes, typically using codes with transversal Clifford gates as their base codes along with different tricks to simulate the transversal action of the non-Clifford gate, could provide smaller overheads. Yet, while avoiding the primary obstacle of concatenation, such schemes may suffer in alternative ways, such as reduced threshold, increased ancilla space, longer measurement gadgets, etc. A study of these alternative methods would be of great interest to the quantum error correction community and would likely involve a new set of tools to analyze the overall overhead.

ACKNOWLEDGMENTS

T.J. would like to acknowledge the support of NSERC and the Vanier-Banting Secretariat through the Vanier CGS. C.C. would like to acknowledge the support of QEII-GSST and to thank S. Weiss for providing the necessary computational resources. This work was supported by CIFAR, NSERC, and Industry Canada.

APPENDIX: ANCILLA PREPARATION AND EXREC CIRCUITS FOR THE 105- AND 49-QUBIT CODES

In order to describe the ancilla states $|\bar{0}\rangle$ and $|\overline{+}\rangle$ for the 105- and 49-qubit codes, we first obtain their encoding circuits for the 7- and 15-qubit codes. Note that the 15-qubit Reed-Muller code is not self-dual which means that the encoding circuit $|\overline{+}\rangle_{15}$ cannot simply be obtained by reversing the direction of each CNOT gate and swapping the physical $|0\rangle$ and $|+\rangle$ states in $|\bar{0}\rangle_{15}$. As will be shown, this asymmetry will result in a larger number of physical locations in $|\overline{+}\rangle_{15}$ compared to $|\bar{0}\rangle_{15}$.

For CSS stabilizer codes, the encoded states can be obtained by solving a partial Latin rectangle using the codes stabilizer generators [39]. However, as was shown in Ref. [19], the circuits obtained from the previous method can be further optimized by considering overlaps in the codes stabilizer generators. As an example, consider the encoding circuit for the state $|\bar{0}\rangle_7$ of the 7-qubit code. The circuit in Fig. 12(a) (obtained via Steane's Latin rectangle method) contains nine CNOT gates. The logical state $|\bar{0}\rangle_7$ can also be obtained by the circuit in Fig. 12(b) which uses eight CNOT gates, one fewer than the previous circuit. To see this, recall that the X stabilizer generators for the 7-qubit code take the form

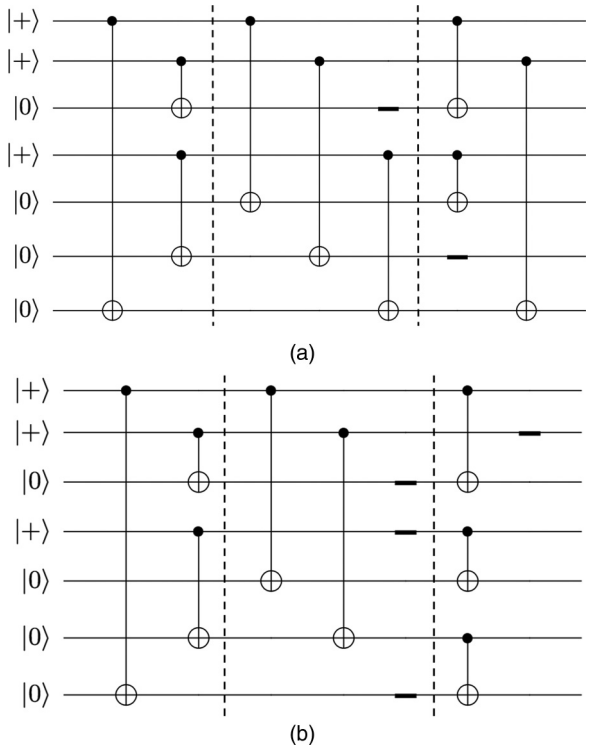

FIG. 12. Encoding of $|\bar{0}\rangle_7$ for the 7-qubit Steane code. The circuit in (a) is obtained using Steane's Latin rectangle method and contains nine CNOT gates. The circuit in (b) was obtained using the stabilizer overlap method and contains eight CNOT gates instead of nine. The dotted vertical lines are used to separate the time steps for which gates are applied in parallel. The bold dark lines represent resting qubits subject to storage errors. Note that we do not include a storage error on the fifth qubit in the first time step since it can be initialized in the second time step.

$g_1 = IIIXXXX$, $g_2 = IXXIIXX$, and $g_3 = XIXIXIX$. In Fig. 12(a), it can be seen that qubit seven is the target of qubits two and four and the corresponding stabilizer generators g_1 and g_2 overlap on qubits six and seven. Consequently, it is possible to replace the two CNOTs with control qubits two and four and target qubit seven with a CNOT having control on the sixth qubit and target on the seventh qubit. We use the circuit in Fig. 12(b) as the outer level of the state $|\bar{0}\rangle_{105}$. Since the 7-qubit code is self-dual, the $|\overline{+}\rangle_7$ can be obtained by reversing the direction of each CNOT gate and swapping the physical $|0\rangle$ and $|+\rangle$ states.

For the 49-qubit code, we use the circuits in Fig. 13 at the outer level in order to minimize the total number of wait times experienced by each qubit. To see this, recall that the logical Z operator for the 15-qubit code can take the form $\overline{Z} = Z_1Z_2Z_3$ (a weight-3 operator, since the first three bits of the codewords have even parity). Consequently, it is possible to apply a logical CNOT gate with the control lying on the first three qubits of a 15-qubit code block and target on a single-qubit code block. This would require the use of only three physical CNOT gates, one per time step. In this way, a Z error on a single-qubit block would propagate to a logical Z error on a 15-qubit code block. Since CNOT gates between 15-qubit code blocks can be implemented transversally, a closer

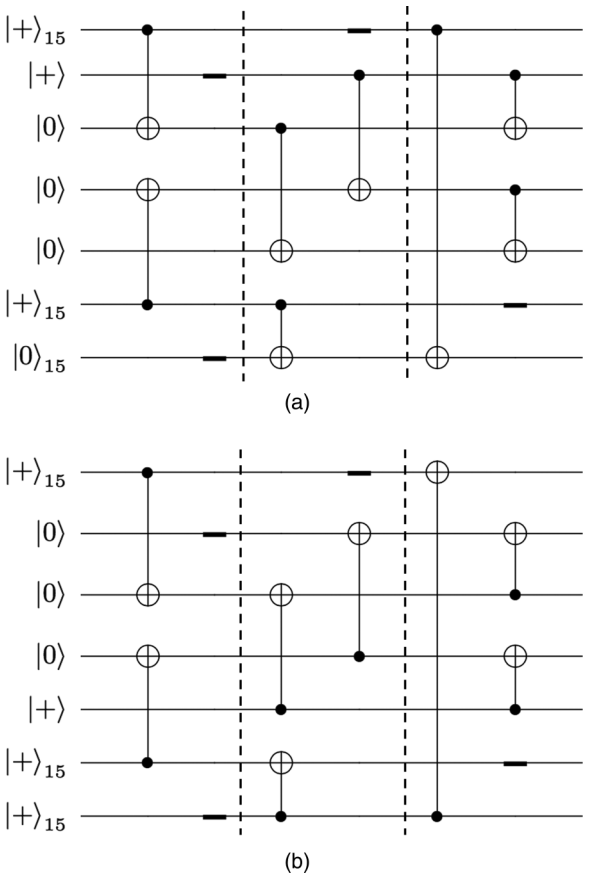

FIG. 13. $|\overline{0}\rangle_{49}$ and $|\overline{+}\rangle_{49}$ circuits for the 49-qubit code. These circuits have a lower spacetime overhead compared to the one used in Fig. 12(b). For example, the CNOT with block 1 as control and block 3 as target in (a) and (b) can be implemented in three time steps instead of seven since $\overline{Z} = Z_1Z_2Z_3$ is a logical Z operator for the 15-qubit Reed-Muller code. Furthermore, any CNOT gate between blocks 1, 6, and 7 can be implemented transversally.

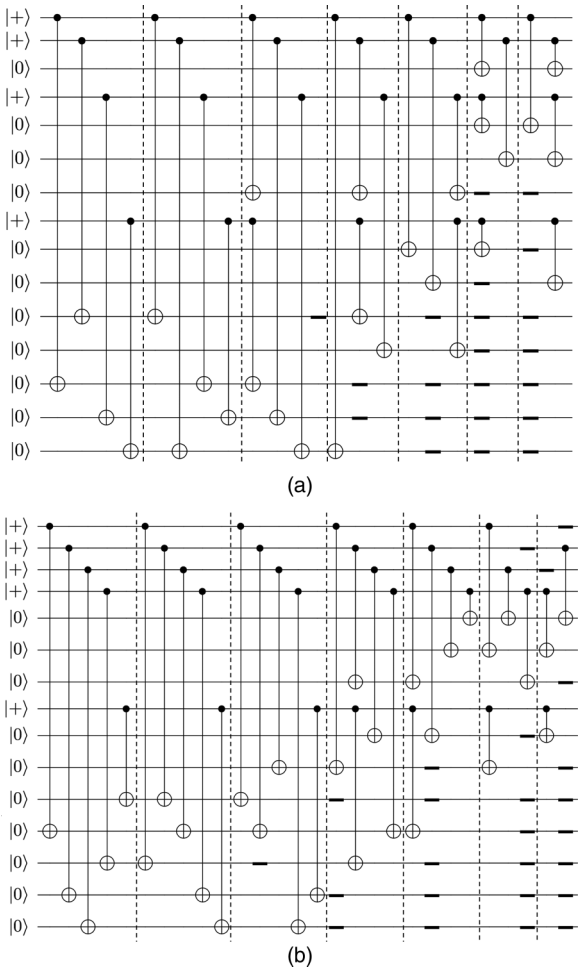

FIG. 14. Encoding circuits $|\overline{0}\rangle_{15}$ and $|\overline{+}\rangle_{15}$ for the 15-qubit Reed-Muller code using Stean's Latin rectangle method. $|\overline{0}\rangle_{15}$ contains 28 CNOT gates and 21 resting qubit locations. $|\overline{+}\rangle_{15}$ contains 32 CNOT gates and 25 resting qubit locations.

look at Fig. 13 shows that the first two CNOT gates can be implemented in three time steps and all other CNOT gates can be implemented in a single time step. Hence, the overall number of time steps required to implement the CNOT gates in Fig. 13 is $3 + 1 + 1 = 5$. If instead we were to use the circuits in Fig. 12, there are three CNOT gates coupling 15-qubit code blocks to single-qubit blocks. Two of these CNOT gates have their control on a 15-qubit code block encoding a $|\overline{+}\rangle$ state and so would be implemented in three time steps each. The CNOT gate coupling the $|+\rangle$ state to the 15-qubit $|\overline{0}\rangle$ code block would require 7 time steps since the minimum weight logical X for the 15-qubit code is 7. The total number of time steps would have been $7 + 3 + 3 = 13$ instead of 5.

Next, we apply the stabilizer overlap method to obtain optimized encoded $|\overline{0}\rangle_{15}$ and $|\overline{+}\rangle_{15}$ states of the 15-qubit Reed-Muller code. Figures 14(a) and 14(b) give the encoding circuits for the $_{15}$ and $|\overline{+}\rangle_{15}$ states using Steane's Latin rectangle method. There are 28 CNOT gates in $|\overline{0}\rangle_{15}$ and 32 CNOT gates in $|\overline{+}\rangle_{15}$. For the circuit $|\overline{0}\rangle_{15}$, the stabilizer overlap method removes six CNOT gates. For the $|\overline{+}\rangle_{15}$, instead of using the stabilizer overlap method to reduce the number of CNOT gates, we consider all pairs of CNOT gates that have the same target qubit. For a given pair, we replace the pair by a single CNOT with a different control qubit. We test all 13 different possible controls until we obtain a correct $|\overline{+}\rangle_{15}$ state. Applying the latter technique, we were able to remove seven CNOT gates from the $|\overline{+}\rangle_{15}$ state [see Figs. 15(a) and 15(b)].

Concatenating the optimized circuits of Fig. 12 with the circuits of Fig. 15 and taking the 7-qubit code as the outer code, we obtain the $|\overline{0}\rangle_{105}$ and $|\overline{+}\rangle_{105}$ ancilla states. The 49-qubit code ancilla states are obtained by concatenating the states of Fig. 13 with those of Fig. 15 (see Table VII for an enumeration of all the types of locations in 49- and 105-qubit ancilla states). A 1-EC contains four $|\overline{0}\rangle$ states, four $|\overline{+}\rangle$ states, eight encoded CNOT gates, two storage locations and four encoded Z-measurement and X-measurement locations (see Fig. 4). Using the values of Table VII, the total number of locations in a 1-EC circuit of the 105-qubit code is thus 7110.

Threshold estimates for the 7-qubit CSS code and the 23-qubit Golay code were calculated in Refs. [18,19]. The threshold calculation was limited by the exrec with the largest number of locations which, in both cases, corresponded to the CNOT-exrec. Given the nontransversal nature of the T gate in the 7-qubit code and the Hadamard gate in the 15-qubit

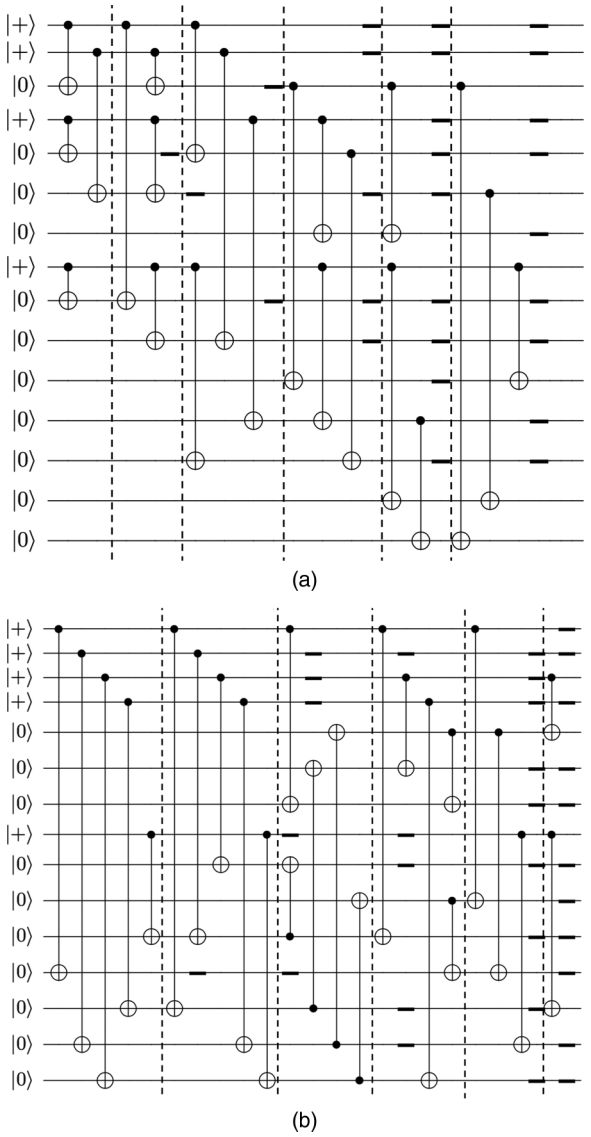

FIG. 15. Optimized encoding circuits for the 15-qubit Reed-Muller code using the stabilizer overlap method for $|\overline{0}\rangle_{15}$ and a computer search algorithm for $|\overline{+}\rangle_{15}$. $|\overline{0}\rangle_{15}$ contains 22 CNOT gates and 27 resting qubit locations. $|\overline{+}\rangle_{15}$ contains 25 CNOT gates and 31 resting qubit locations.

code, the threshold calculation is no longer limited by the exrec with the largest number of locations. As was shown in Sec. IV D, the threshold calculation is instead limited by

TABLE VII. Enumeration for all the types of locations for the $|\overline{0}\rangle_{105}$, $|\overline{+}\rangle_{105}$, $|\overline{0}\rangle_{49}$, and $|\overline{+}\rangle_{49}$ states.

	$\|\overline{0}\rangle_{105}$	$\|\overline{+}\rangle_{105}$	$\|\overline{0}\rangle_{49}$	$\|\overline{+}\rangle_{49}$
Number of CNOTs	298	301	112	115
Number of resting qubits	246	250	248	252
Number of physical $\|0\rangle$ states	74	73	34	33
Number of physical $\|+\rangle$ states	31	32	15	16
Total	649	656	409	416

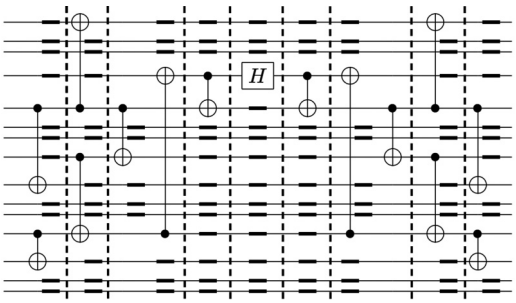

FIG. 16. Logical Hadamard H circuit for [[15,1,3]] Reed-Muller code. Logical H for the 105-qubit code is implemented fault tolerantly by applying each non-fault-tolerant logical H gates in parallel. Note that there are a total of nine time steps. Consequently, the 49-qubit code Hadamard circuit will contain a single physical Hadamard and 8 resting qubit locations on blocks 2–5.

the Hadamard-exrec. It is thus worthwhile to analyze the Hadamard circuit in more detail.

The encoding circuit for the Hadamard gate must map the stabilizer generators of the 15-qubit code to an element of the stabilizer group. Furthermore, since $HXH = Z$ and $HZH = X$, the logical X operator of the 15-qubit code must be mapped to the logical Z operator and vice versa. The circuit in Fig. 16 satisfies these properties. To derive such a circuit, we wrote a program in MATLABwhich inserted CNOT gates at random locations within the 15 physical qubits and propagated all stabilizer generators and logical operators through the circuit. If all operators transformed appropriately as described above, then the locations of the CNOT gates were recorded. The best circuit that we found contains a total of 14 CNOT gates, 1 physical Hadamard gate, and has a depth of 9 time steps (see Table VIII for a complete enumeration of the locations in the Hadamard circuit for the 15-qubit, 49-qubit, and 105-qubit codes). However, it is still an open question as to whether a circuit using fewer CNOT gates with a smaller depth can be found.

One important aspect of the circuit in Fig. 16 is that input Z errors from the LEC are much more likely to lead to a logical error at the output of the Hadamard circuit than input X errors. To illustrate this, we consider two different cases. In the first case, we insert a single X error at the input of the Hadamard circuit and propagate the error throughout the circuit to determine if a logical error occurred at the output

TABLE VIII. Enumeration of all the types of locations for the 15-, 49-, and 105-qubit Hadamard gates. Restricting to the 105-qubit code, since a 1-EC circuit contains 7110 locations, the total number of locations in the Hadamard-exrec is 15 067. These quantities will be relevant in the adversarial noise threshold calculation of the Hadamard-exrec as well as in the resource overhead calculation.

	H_{15}	H_{49}	H_{105}
Number of CNOTs	14	42	98
Number of resting qubits	106	350	742
Number of physical H states	1	7	7
Total	121	399	847

of the circuit, which occurs in the case of qubits 4, 8, and 12. In the second case, we perform the exact same operations but with a single Z error. In this case, qubits 1, 4, 5, 8, 9, 12, and 13 produce different logical faults at the output of the Hadamard circuit compared to when input X errors were considered. It should then be expected that Z errors will play an important role when calculating the noise threshold for the concatenated scheme.

For the 105-qubit code, logical Hadamard is obtained by applying the nontransversal logical Hadamard of Fig. 16 to each of the encoded 15-qubit code blocks. For the 49-qubit code, the logical Hadamard gate is obtained by applying the nontransversal logical Hadamard of Fig. 16 to blocks 1, 6, and 7. Blocks 2–5 will consist of a physical Hadamard gate along with 8 resting qubit locations (since there are a total of 9 time steps in Fig 16).

As was explained above, a single error can lead to a logical fault when propagating through the 15-qubit Hadamard circuit. However, since the Hadamard circuit is transversal for the 7-qubit CSS code, a weight-1 error will lead to at most a *single* logical fault on one of the code blocks which will be corrected by the outer code. Thus, the Hadamard circuit for the 49- and 105-qubit code is fault tolerant.

Next, we analyze the construction of the T-gate circuit for the 7-qubit code following the ideas of Refs. [20,21]. Similarly to the Hadamard circuit, to obtain a circuit that correctly encodes the T gate for the 7-qubit code, all stabilizer generators must be mapped to elements of the stabilizer group. Furthermore, since $TXT^\dagger = \frac{1}{\sqrt{2}}(X+Y)$ and $TZT^\dagger = Z$, we require that the logical operators for the 7-qubit code transform as $\overline{X} \to \overline{X}(I+\overline{Z})$ and $\overline{Z} \to \overline{Z}$. The circuit in Fig. 17 satisfies the above properties. It should be noted that regardless of the error model studied, we treated

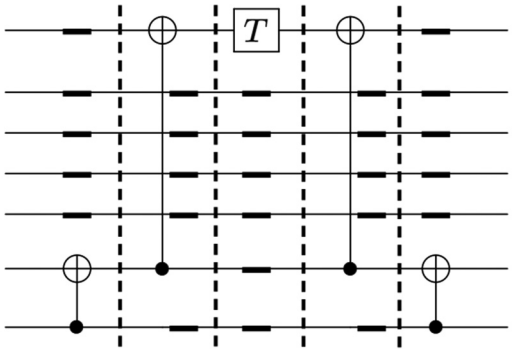

FIG. 17. Logical T circuit for [[7,1,3]] CSS code. All stabilizer generators map to elements in the stabilizer group. Furthermore, $\overline{X} \to \overline{X}(I+\overline{Z})$ and $\overline{Z} \to \overline{Z}$ as required.

the transformation of X errors in our simulations by taking an adversarial approach to their transformation to X or Y errors, depending on the other errors in the circuit. That is, we considered both the case when the X error transformed to either X or Y, and treated the worst case logical error outcome based on the choice of error. This sufficed to prove a lower bound on the appropriate threshold.

Logical T for the 49- and 105-qubit codes is constructed from the circuit in Fig. 17 with each code block encoded using the 15-qubit Reed-Muller code (for the 49-qubit code, blocks 2–5 only contain one qubit). The construction is not fault tolerant on the outer code since errors can spread between code blocks. However, since the underlying logical gates are transversal on the 15-qubit code blocks, a single error would propagate to at most a single error on each code block which would be corrected by the inner code.

[1] Y. S. Weinstein, C. S. Hellberg, and J. Levy, Quantum-dot cluster-state computing with encoded qubits, Phys. Rev. A **72**, 020304 (2005).

[2] R. Stock and D. F. V. James, Scalable, High-Speed Measurement-Based Quantum Computer Using Trapped Ions, Phys. Rev. Lett. **102**, 170501 (2009).

[3] S. J. Devitt, A. G. Fowler, A. M. Stephens, A. D. Greentree, L. C. Hollenberg, W. J. Munro, and K. Nemoto, Architectural design for a topological cluster state quantum computer, New J. Phys. **11**, 083032 (2009).

[4] R. Van Meter, T. D. Ladd, A. G. Fowler, and Y. Yamamoto, Distributed quantum computation architecture using semiconductor nanophotonics, International, J. Quantum. Inform. **08**, 295 (2010).

[5] N. C. Jones, R. Van Meter, A. G. Fowler, L. McMahon, J. Kim, T. D. Ladd, and Y. Yamamoto, Layered Architecture for Quantum Computing, Phys. Rev. X **2**, 031007 (2012).

[6] N. Y. Yao, L. Jiang, A. V. Gorshkov, P. C. Maurer, G. Giedke, J. I. Cirac, and M. D. Lukin, Scalable architecture for a room temperature solid-state quantum information processor, Nat. Commun. **3**, 800 (2012).

[7] M. H. Devoret and R. J. Schoelkopf, Superconducting circuits for quantum information: an outlook, Science **339**, 1169 (2013).

[8] S. J. Devitt, A. M. Stephens, W. J. Munro, and K. Nemoto, Requirements for fault-tolerant factoring on an atom-optics quantum computer, Nat. Commun. **4**, 2524 (2013).

[9] R. Barends, J. Kelly, A. Megrant, A. Veitia, D. Sank, E. Jeffrey, T. C. White, J. Mutus, A. G. Fowler, B. Campbell *et al.*, Superconducting quantum circuits at the surface code threshold for fault tolerance, Nature (London) **508**, 500 (2014).

[10] J. M. Chow, J. M. Gambetta, E. Magesan, D. W. Abraham, A. W. Cross, B. R. Johnson, N. A. Masluk, C. A. Ryan, J. A. Smolin, S. J. Srinivasan *et al.*, Implementing a strand of a scalable fault-tolerant quantum computing fabric, Nat. Commun. **5**, 4015 (2014).

[11] J. Kelly, R. Barends, A. G. Fowler, A. Megrant, E. Jeffrey, T. C. White, D. Sank, J. Y. Mutus, B. Campbell, Y. Chen *et al.*, State preservation by repetitive error detection in a superconducting quantum circuit, Nature (London) **519**, 66 (2015).

[12] M. Takita, A. D. Córcoles, E. Magesan, B. Abdo, M. Brink, A. Cross, J. M. Chow, and J. M. Gambetta, Demonstration

of Weight-Four Parity Measurements in the Surface Code Architecture, Phys. Rev. Lett. **117**, 210505 (2016).

[13] D. Aharonov and M. Ben-Or, Fault-tolerant Quantum Computation with Constant Error, in *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, Vol. 176 (ACM, New York, 1997).

[14] J. Preskill, Reliable quantum computers, Proc. R. Soc. London, Ser. A **454**, 385 (1998).

[15] A. Y. Kitaev, Quantum computations: algorithms and error correction, Russ. Math. Surveys **52**, 1191 (1997).

[16] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, Topological quantum memory, J. Math. Phys. **43**, 4452 (2002).

[17] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Surface codes: Towards practical large-scale quantum computation, Phys. Rev. A **86**, 032324 (2012).

[18] P. Aliferis, D. Gottesman, and J. Preskill, Quantum accuracy threshold for concatenated distance–3 codes, Quantum Inf. Comput. **6**, 97 (2006).

[19] A. Paetznick and B. W. Reichardt, Fault-tolerant ancilla preparation and noise threshold lower bounds for the 23-qubit golay code, Quantum Inf. Comput. **12**, 1034 (2011).

[20] T. Jochym-O'Connor and R. Laflamme, Using Concatenated Quantum Codes For Universal Fault-Tolerant Quantum Gates, Phys. Rev. Lett. **112**, 010505 (2014).

[21] C. Chamberland, T. Jochym-O'Connor, and R. Laflamme, Thresholds for Universal Concatenated Quantum Codes, Phys. Rev. Lett. **117**, 010501 (2016).

[22] B. Eastin and E. Knill, Restrictions on Transversal Encoded Quantum Gate Sets, Phys. Rev. Lett. **102**, 110502 (2009).

[23] B. Zeng, A. W. Cross, and I. L. Chuang, Transversality versus universality for additive quantum codes, IEEE Trans. Inf. Theory **57**, 6272 (2011).

[24] F. Pastawski and B. Yoshida, Fault-tolerant logical gates in quantum error-correcting codes, Phys. Rev. A **91**, 012305 (2015).

[25] A. M. Steane, Active Stabilization, Quantum Computation, and Quantum State Synthesis, Phys. Rev. Lett. **78**, 2252 (1997).

[26] H. Häffner, W. Hänsel, C. Roos, J. Benhelm, M. Chwalla, T. Körber, U. Rapol, M. Riebe, P. Schmidt, C. Becher *et al.*, Scalable multiparticle entanglement of trapped ions, Nature (London) **438**, 643 (2005).

[27] E. Nikahd, M. Sedighi, and M. S. Zamani, Non-uniform code concatenation for universal fault-tolerant quantum computing, arXiv:1605.07007.

[28] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, England, 2000).

[29] J. T. Anderson and G. Duclos-Cianci (private communication).

[30] P. P. Aliferis, Level reduction and the quantum threshold theorem. Ph.D. thesis, California Institute of Technology, 2007.

[31] S. Bravyi and A. Kitaev, Universal quantum computation with ideal Clifford gates and noisy ancillas, Phys. Rev. A **71**, 022316 (2005).

[32] A. M. Meier, B. Eastin, and E. Knill, Magic-state distillation with the four-qubit code, Quantum Inf. Compute. **13**, 195 (2013).

[33] A. Paetznick and B. W. Reichardt, Universal Fault-Tolerant Quantum Computation with Only Transversal Gates and Error Correction, Phys. Rev. Lett. **111**, 090505 (2013).

[34] J. T. Anderson, G. Duclos-Cianci, and D. Poulin, Fault-Tolerant Conversion between the Steane and Reed-Muller Quantum Codes, Phys. Rev. Lett. **113**, 080501 (2014).

[35] S. Bravyi and A. Cross, Doubled color codes, arXiv:1509.03239.

[36] T. Jochym-O'Connor and S. D. Bartlett, Stacked codes: Universal fault-tolerant quantum computation in a two-dimensional layout, Phys. Rev. A **93**, 022323 (2016).

[37] C. Jones, P. Brooks, and J. Harrington, Gauge color codes in two dimensions, Phys. Rev. A **93**, 052332 (2016).

[38] T. J. Yoder, R. Takagi, and I. L. Chuang, Universal Fault-Tolerant Gates on Nondegenerate Stabilizer Codes, Phys. Rev. X **6**, 031039 (2016).

[39] A. M. Steane, Fast fault-tolerant filtering of quantum codewords, arXiv:quant-ph/0202036.